

# SIMULACIÓ DE L'ACTUACIÓ DISTRIBUÏDA DE ROBOTS MODULARS

Óscar Rodríguez Pestaña

*Gener 2013*



**Directora:** Vera Sacristán Adinolfi

**Titulació:** Enginyeria Tècnica en Informàtica de Gestió

**Universitat:** Universitat Politècnica de Catalunya (UPC)

**Centre:** Facultat d'Informàtica de Barcelona (FIB)

**Departament:** Departament de Matemàtica Aplicada II



# Continguts

<b>1. Introducció .....</b>	<b>1</b>
1.1 Robots modulars .....	1
1.2 Model del projecte .....	3
1.3 Objectiu del projecte .....	7
1.4 Estructura de la memòria .....	8
<b>2. Cerca del màster .....</b>	<b>9</b>
2.1 Objectiu .....	9
2.2 Estratègia .....	9
2.3 Adaptació .....	10
2.4 Problemes .....	11
2.4.1 Prioritats .....	11
2.4.2 Estats intermedis .....	12
2.4.3 Canals de comunicació .....	12
2.4.4 Visualització de missatges .....	12
2.4.5 Solapament de colors .....	13
2.5 Modificacions .....	13
2.6 Models de prova .....	14
2.6.1 Solapament de missatges de la mateixa fase .....	14
2.6.2 Solapament de missatges de fases diferents .....	15
2.6.3 Creuament de missatges de la mateixa fase .....	15
2.6.4 Creuament de missatges de fases diferents .....	16
<b>3. Detecció de forats .....</b>	<b>17</b>
3.1 Objectiu .....	17
3.2 Estratègia .....	17

3.3	Adaptació .....	18
3.4	Problemes .....	19
3.4.1	Prioritats .....	19
3.4.2	Estats intermedis .....	19
3.4.3	Canals de comunicació .....	20
3.4.4	Visualització de missatges .....	20
3.4.5	Solapament de colors .....	20
3.5	Modificacions .....	20
3.6	Models de prova .....	21
3.6.1	Solapament de missatges de la mateixa fase .....	21
3.6.2	Solapament de missatges de fases diferents .....	22
3.6.3	Creuament de missatges .....	22
<b>4.</b>	<b>Construcció de l'<i>scan tree</i> .....</b>	<b>23</b>
4.1	Objectiu .....	23
4.2	Estratègia .....	23
4.3	Regles .....	25
4.4	Problemes .....	25
4.5	Modificacions .....	25
4.6	Alternatives .....	26
4.7	Models de prova .....	26
4.7.1	Forma de cuc .....	26
4.7.2	Forma densa .....	27
4.7.3	Forma semi-densa .....	27
4.7.4	Forat simple .....	27
4.7.5	Forats múltiples .....	27
<b>5.</b>	<b>Construcció de la funció potencial .....</b>	<b>29</b>
5.1	Objectiu .....	29
5.2	Estratègia .....	29

5.3 Regles .....	32
5.3.1 Grup “start” .....	32
5.3.2 Grup “first go” .....	33
5.3.3 Grup “first wait” .....	33
5.3.4 Grup “wait-continue” .....	34
5.3.5 Grup “continue” .....	34
5.3.6 Grup “last” .....	35
5.3.7 Grup “last father” .....	35
5.3.8 Grup extra “first go adapt” i “first wait adapt” .....	36
5.4 Problemes .....	36
5.4.1 Ordre dels fills d’un mòdul .....	36
5.4.2 Repetició del recorregut .....	36
5.4.3 Conflicte d’estats .....	37
5.4.4 Marcatge d’un fill .....	37
5.4.5 Informació extra, identificació de branques .....	37
5.4.6 Regles de “continue” que no apliquen mai .....	37
5.4.7 La funció OR no existeix .....	38
5.4.8 Creuament de fases .....	38
5.4.9 Missatge de marca .....	38
5.4.10 Conflictes d’adaptació amb les regles de <i>Master</i> i <i>Hole</i> .....	39
5.5 Modificacions .....	39
5.6 Alternatives .....	40
5.7 Models de prova .....	41
5.7.1 Propagació del missatge d’anada .....	41
5.7.2 Propagació del missatge de tornada .....	42
5.7.3 Propagació del missatge a una fulla .....	42
5.7.4 Bifurcació per la primera branca .....	43
5.7.5 Bifurcació per l’última branca .....	43
5.7.6 Tornada d’una bifurcació .....	44
<b>6. Reconfiguració a la forma canònica .....</b>	<b>45</b>
6.1 Objectiu .....	45
6.2 Estratègia .....	45
6.2.1 Activació .....	45
6.2.2 Moviment .....	46
6.2.3 Adhesió .....	51
6.2.4 Exemple global de l’estratègia .....	51

6.3	Regles .....	53
6.3.1	Grup "activate" .....	54
6.3.2	Grup "slide" .....	55
6.3.3	Grup "convex transition" .....	56
6.3.4	Grup "concave transition" .....	57
6.3.5	Grup "opposite transition" .....	58
6.3.6	Grup "strip" .....	60
6.4	Problemes .....	60
6.4.1	Informació errònia d'un mòdul .....	60
6.4.2	Comunicació d'intencionalitats .....	61
6.4.3	Obstruccions .....	61
6.4.4	Identificar un coll d'ampolla .....	62
6.4.5	Formes de les branques .....	62
6.4.6	Informació extra, identificació de branques .....	62
6.4.7	Informació extra, comparació de comptadors de 2 mòduls .....	63
6.4.8	Caminar sobre la tirallonga .....	63
6.4.9	Identificació de l'últim mòdul .....	63
6.4.10	Simulador .....	63
6.5	Modificacions .....	64
6.6	Alternatives .....	65
6.7	Models de prova .....	66
6.7.1	Noves funcionalitats del simulador .....	67
6.7.2	Colls d'ampolla .....	68
6.7.3	Colls d'ampolla (cas especial) .....	68
6.7.4	Col·lisions (comunicació entre mòduls) .....	70
6.7.5	Marcatge de l'últim mòdul .....	71
<b>7.</b>	<b>Construcció de la forma final .....</b>	<b>73</b>
7.1	Objectiu .....	73
7.2	Estratègia .....	73
7.2.1	Obtenció de la informació .....	73
7.2.2	Construcció .....	77
7.3	Regles .....	80
7.3.1	Grup "reverse rules" .....	82
7.3.2	Grup "rollback" .....	82
7.3.3	Grup "get info" .....	82

7.4 Problemes .....	83
7.4.1 Forats .....	83
7.4.2 Forat “especial” .....	84
7.4.3 Cul de sac “especial” .....	84
7.4.4 Col·lisions .....	84
7.4.5 Obstruccions .....	85
7.4.6 Informació del màster .....	85
7.5 Modificacions .....	85
7.6 Alternatives .....	85
7.7 Models de prova .....	86
7.7.1 Responsable de forat .....	86
7.7.2 Forat “especial” .....	87
7.7.3 Rollback .....	87
7.7.4 Cul de sac .....	89
<b>8. Simulador .....</b>	<b>91</b>
8.1 Descripció .....	91
8.2 Funcionalitats .....	93
8.3 Modificacions .....	95
8.3.1 Noves funcionalitats: comparació numèrica a distància .....	95
8.3.2 Ampliació de funcionalitat: càlculs amb dades externes .....	96
8.3.3 Ampliació de canals: nous canals de comunicació .....	97
8.4 A mode de llegenda .....	98
<b>9. Gestió del projecte .....</b>	<b>101</b>
9.1 Planificació i execució .....	101
9.2 Pressupost .....	109
<b>10. Conclusions .....</b>	<b>111</b>
10.1 Resultats obtinguts .....	111
10.2 Dificultats trobades .....	111

10.3 Futur del projecte .....	112
10.4 Valoració personal .....	113
<b>Referències .....</b>	<b>115</b>
<b>Annexos .....</b>	<b>119</b>
A. Regles de la fase de construcció de l' <i>scan tree</i> .....	125
B. Regles de la fase de construcció de la funció potencial .....	131
C. Regles de la fase de reconfiguració a la forma canònica .....	217
D. Regles de la fase de construcció de la forma final .....	333



# 1 Introducció

Aquest projecte estudia la simulació de l'actuació de robots modulars de forma distribuïda. Així doncs en aquest capítol expliquem què és un robot modular, les diferències amb altres tipus de robots i alguns dels models que ens podem trobar. Expliquem quines són les característiques del model utilitzat en el projecte i les capacitats del sistema que fem servir per realitzar la simulació. Finalment descrivim els objectius del projecte i l'organització de la memòria.

## 1.1 Robots modulars

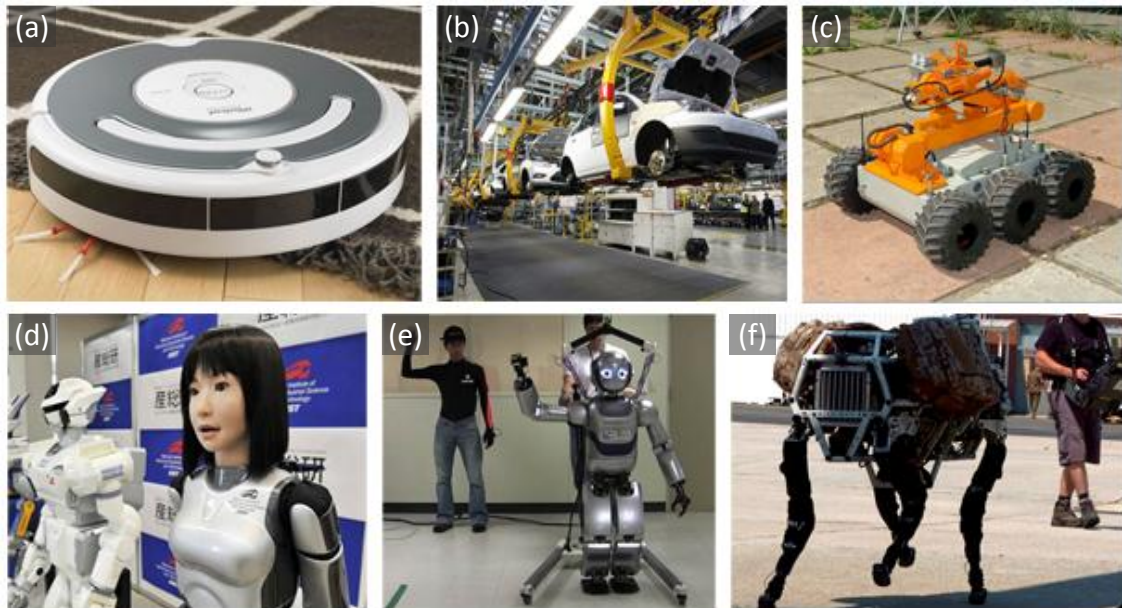
Un robot és un dispositiu (generalment mecànic) que realitza tasques automàticament, ja sigui amb supervisió humana, a través d'un programa predefinit o seguint un conjunt de regles utilitzant tècniques d'intel·ligència artificial. Normalment la seva funció és substituir o ajudar l'humà en tasques complexes o perilloses (vegeu-ne un exemple a la Figura 1.1).



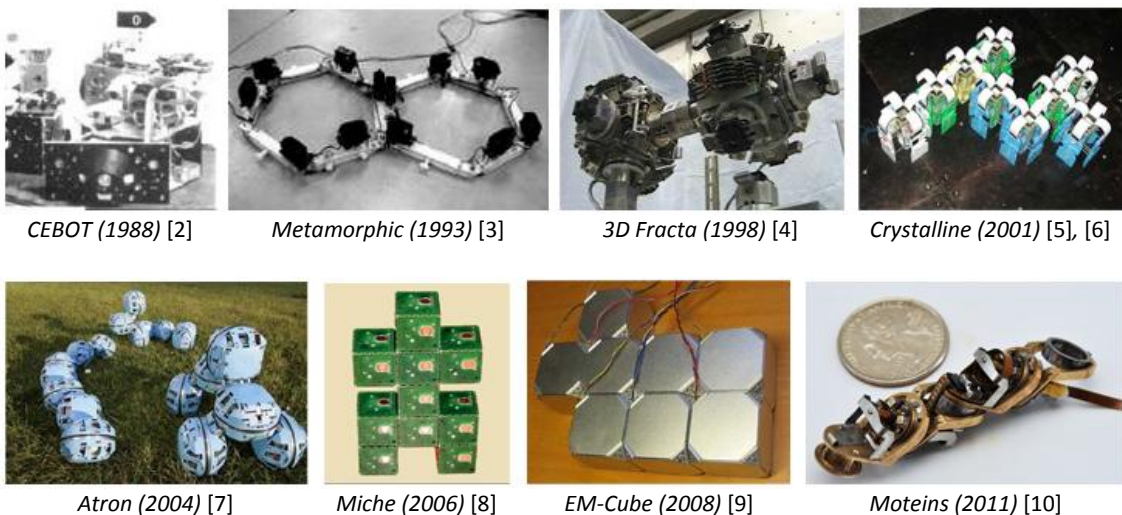
Figura 1.1: Robot desactivador de bombes.

Hi ha molts tipus diferents de robots. Els més comuns actualment són els robots especialitzats, que serveixen per ajudar a una tasca concreta. Aquests robots poden ser tant a nivell d'usuari (un robot aspirador), com a nivell industrial (una cadena de muntatge). A la Figura 1.2 podeu veure una varietat de robots diferents.

Una de les branques d'estudi actual de la robòtica es centra en el desenvolupament dels robots modulars. La idea és aconseguir que una gran quantitat de petits robots puguin interactuar entre ells per dur a terme una tasca conjunta.

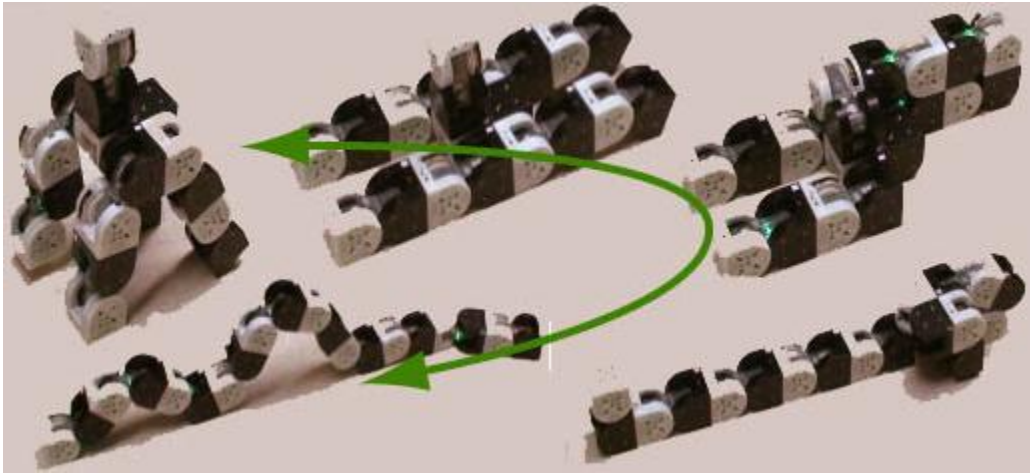


L'origen d'aquest tipus de robot modular es remunta a 1988 amb la proposta de Toshio Fukuda i el seu robot CEBOT (*CELLular roBOT*) considerant cada mòdul com una cèl·lula autònoma que pot moure's i connectar-se a unes altres. A partir de llavors s'han anat introduint modificacions o alternatives com ara manipuladors metamòrfics per acoblar-se o desacoblar-se uns mòduls amb altres, la construcció d'estructures en tres dimensions, el nombre d'articulacions o cares d'un mòdul, etcètera [1]. Tot i que són força recents, a la Figura 1.3 podeu veure com han anat canviant amb el temps.



**Figura 1.3: Evolució dels robots modulars.** Alguns dels robots modulars que s'han anat creant des de que va aparèixer el primer en 1988.

Aquesta forma d'interactuar uns mòduls amb altres pot fer que un robot s'adapti al seu entorn per aconseguir realitzar la tasca assignada. Per posar un exemple, si tenim un robot amb una càmera incorporada però necessitem que busqui supervivents entre restes d'un terratrèmol, un robot mòbil potser és capaç de recórrer una superfície inestable per no posar en perill més persones, però un robot modular podria ser capaç de modificar la seva forma (vegeu la Figura 1.4) de manera que pogués colar-se entre les runes o pujar per llocs inaccessibles per a una persona i identificar si hi ha algú sota les restes per centrar la cerca sobre aquella zona.

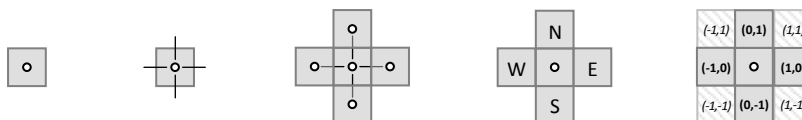


**Figura 1.4: Robot MTRAN-III (2005).** Es mostren diferents formes que pot adoptar el robot [11].

Una altra característica d'aquest tipus de robots és que tots els mòduls que formen part d'un robot són iguals. Això dóna lloc a una altra de les seves avantatges que és la capacitat d'autoreparació. En el cas que un mòdul quedi inoperatiu, un altre mòdul el pot substituir. Això és essencial per tasques on una persona no pot accedir fàcilment per reparar o substituir el robot com per exemple en missions a l'espai.

## 1.2 Model del projecte

Al nostre projecte s'aplica als models de robots reticulars que estan formats per mòduls quadrats de dues dimensions. Cada mòdul té la possibilitat de connectar-se i desconnectar-se en les 4 direccions (N, S, E, W). Podeu veure'n una representació a la Figura 1.5.



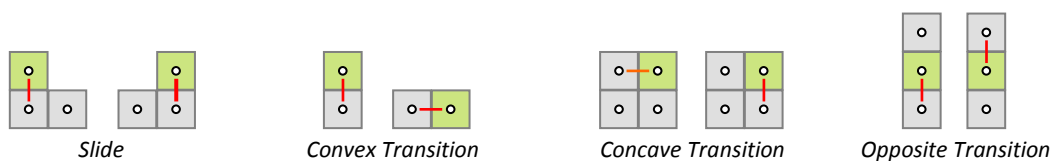
**Figura 1.5: Representació dels mòduls d'un robot.** Es mostra la representació dels mòduls que utilitzem al llarg del projecte. D'esquerra a dreta: mòdul desconnectat, mòdul amb les connexions actives, mòdul connectat a tots els veïns, punts cardinals de referència per a un mòdul, i coordenades relatives de referència per un mòdul.

Al llarg de la memòria del projecte fem servir colors per representar l'estat d'alguns dels mòduls. Molts colors representen la fase en la qual es troben els mòduls per facilitar la seva comprensió, aquest color és el mateix color identificatiu utilitzat en el simulador. Hi ha 2 mòduls especials que tenen colors reservats (Figura 1.6).



**Figura 1.6: Colors identificatius dels mòduls.**

Cadascun dels mòduls pot realitzar 4 tipus de moviments: dos de desplaçament, l'*Slide* i el *Convex Transition*; i dos de connexió, el *Concave Transition* i l'*Opposite Transition* (Figura 1.7).



**Figura 1.7: Moviments possibles dels mòduls.** Per cada moviment es mostra a l'esquerra la situació prèvia i a la dreta la situació posterior.

Tots els mòduls són iguals, tenen les mateixes capacitats i apliquen les mateixes regles. Cada mòdul té una capacitat de procés simple i una memòria molt limitada.

Els mòduls utilitzats en el projecte disposen de certes capacitats: cada mòdul pot conèixer la seva orientació (N, S, E, W); detectar si està connectat a un veí; enviar i rebre petits missatges entre veïns; i fer operacions elementals amb comptadors i *strings*.

Cada mòdul té informació pròpia que pot emmagatzemar i modificar: la seva posició relativa; les connexions amb els veïns; l'estat en el qual es troba; comptadors per emmagatzemar valors; i missatges d'entrada i sortida.

Les operacions elementals que pot realitzar un mòdul són les de comparar i combinar dades de comptadors, missatges i nombres enters. A més pot testejar la congruència dels estats i de missatges de text.

L'algorisme de reconfiguració és totalment distribuït.

Cada mòdul actua pel seu compte, sense la necessitat d'un controlador central. El senyal d'inici pot ser enviada a tots els mòduls, però no és necessari, es pot enviar a un de qualsevol per tal que aquest activi la resta. Tots els mòduls actuen en paral·lel i apliquen les regles de forma sincronitzada. El sistema està pensat per ser sincronitzat (una versió asíncrona faria servir "handshaking").



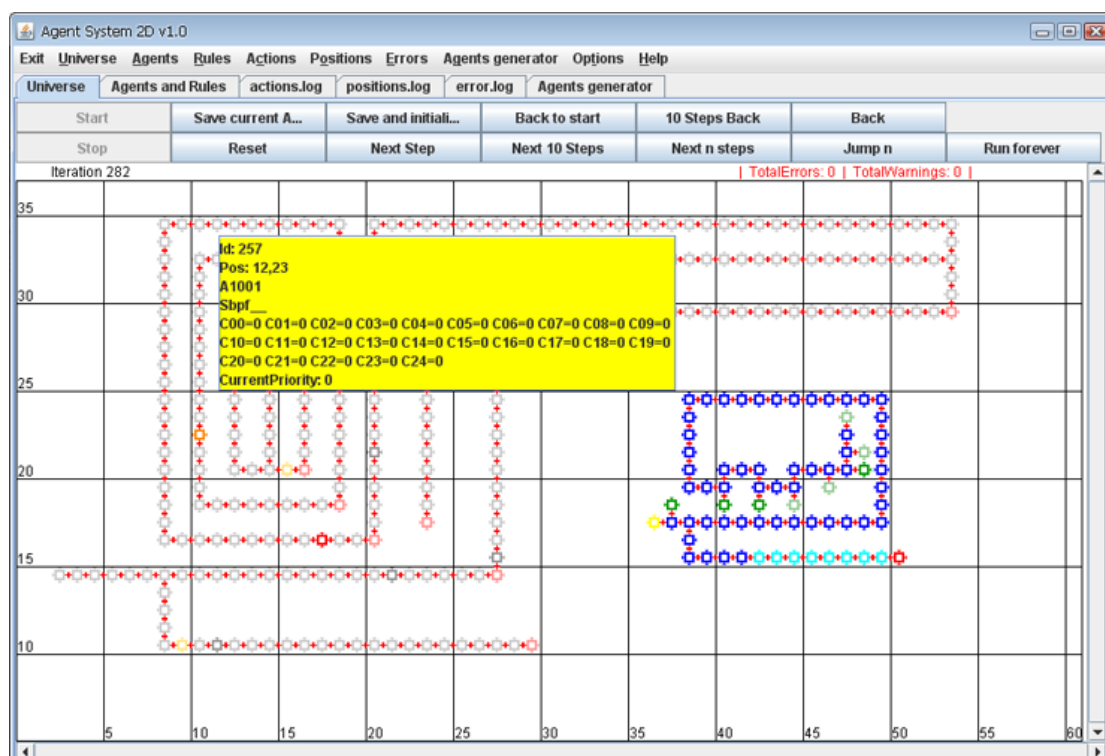
En el nostre sistema, la forma inicial no necessita ser coneguda pels mòduls i la forma final només és necessari que la conegui el "master". El màster és l'encarregat de calcular com anar de la forma canònica a la final.

Aquest model i aquesta estratègia es basen en els treballs previs [12] i [13].

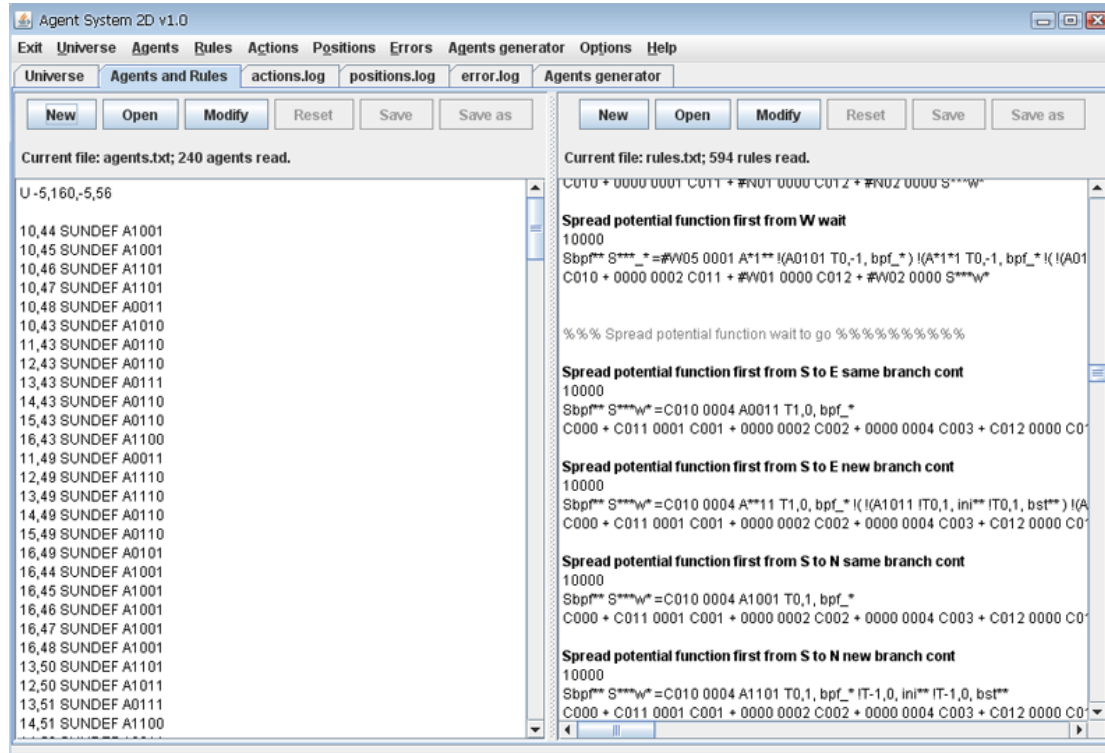
Per altra banda tenim el simulador, l'eina que ens facilita aplicar regles a diferents configuracions. Aquest simulador fa possible treballar amb mòduls que tenen totes les funcions i capacitats esmentades anteriorment.

Les figures 1.8 i 1.9 mostren dues vistes del simulador. En la primera ("Universe") el simulador mostra l'evolució de la reconfiguració del robot i permet manipular les iteracions per veure com afecten les regles al nostre robot en cada pas. En la segona ("Agents and Rules") el simulador ens permet introduir i modificar els mòduls i les regles i guardar-ho tot en fitxers, també permet carregar fitxers.

Altres facilitats que atorga el simulador són: "Actions" que mostra les regles executades per cada mòdul en cada iteració; "Positions" on es mostra la informació de cada mòdul en cada iteració; "Error" on es descriuen els errors detectats en cada iteració; i "Agents generator" una eina que permet generar configuracions de robots de forma fàcil.



**Figura 1.8: Simulador Agent System 2D (Universe).** A Universe podem observar les configuracions dels robots en funcionament, la iteració en la qual ens trobem, el nombre d'errors produïts i informació concreta de qualsevol dels mòduls. També ens permet manipular les iteracions.



**Figura 1.9: Simulador Agent System 2D (Agents and Rules).** A *Agents and Rules* podem gestionar els mòduls i les regles. A l'esquerra estan les posicions i estats inicials dels mòduls. A la dreta es troben les regles que han d'aplicar els mòduls.

La visualització dels robots es realitza sobre una quadrícula en la qual s'observen els canvis que realitzen els mòduls, ja siguin canvis de posició, de connexions o d'estat. El simulador disposa d'eines per facilitar la feina: permet generar els robots de forma fàcil, verificar la correctesa del format de les regles i avisa d'errors durant l'execució, permet diferenciar visualment mòduls per mitjà de colors assignats als seus estats i verificar la informació que tenen en cada iteració, avançar i retrocedir en el temps, etcètera.

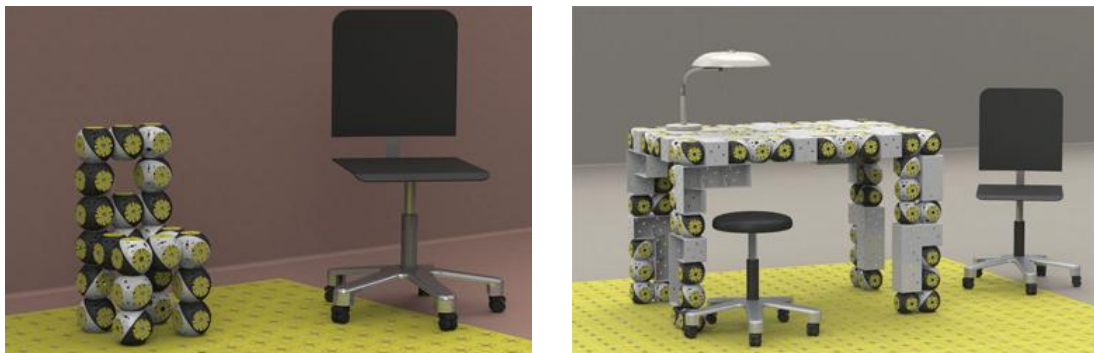
Aquest simulador va ser desenvolupat a [14].

## 1.3 Objectiu del projecte

L'objectiu del nostre projecte és simular com s'han de comportar els mòduls del nostre robot per aconseguir modificar la seva forma.

En el nostre projecte estudiem com aconseguir que una quantitat indeterminada de mòduls ja connectats puguin interactuar entre ells de tal manera que modifiquin la forma del robot per transformar-la en una forma canònica.

Un exemple de la nostra idea extrapolat a 3 dimensions permetria aconseguir de forma automàtica l'objectiu de Biorobotics amb RoomBot (Figura 1.10), reconfigurar els mòduls per formar diferents mobles en funció de les necessitats de l'usuari [15].

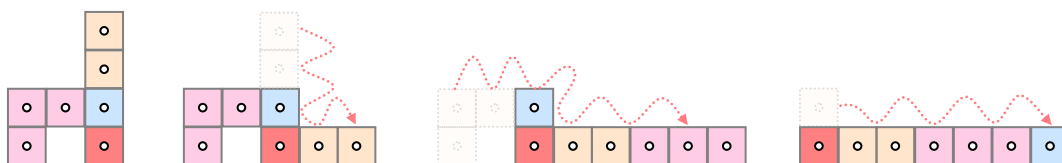


**Figura 1.10: Robot modular RoomBot (2009).** Es mostren dos exemples renderitzats de dues formes diferents generades amb mòduls RoomBots.

La nostra solució es basa en construir un arbre generador del robot de forma que les fulles siguin les úniques que es poden moure. El moviment dels mòduls es realitza al voltant dels límits del robot i sempre segueix la mateixa regla (o la regla de la mà dreta, o la de l'esquerra, sempre la mateixa).

Els moviments dels mòduls no tenen permès anar sobre altres mòduls en moviment (per evitar col·lisions imprevisibles) i la funció potencial la utilitzem per guiar els moviments dels mòduls.

Amb aquest model de robot i aquest sistema hem d'aconseguir desfer la forma inicial per convertir-la en una de nova. La idea conceptual la podeu veure a la Figura 1.11.



**Figura 1.11: Reconfiguració del robot.** Es mostra de forma general el procés que segueix el robot per anar desfent branques per canviar a la forma canònica.

## **1.4 Estructura de la memòria**

Hem dividit la memòria en 10 capítols. En el primer capítol (en el qual ens trobem) hem fet una introducció explicant el context del projecte i les característiques del model de robot modular que utilitzem, així com els objectius generals del projecte i l'estructura d'aquest document.

Els sis capítols següents (del capítol 2 al 7) formen la part principal, cadascun d'ells dedicat a una de les fases del projecte. Els capítols segon i tercer pertanyen a la fase inicial, en un tractem la fase de la cerca del màster, i en l'altre la dels líders de forat. En el quart capítol expliquem el procés de construcció de l'arbre generador. En el cinquè, com realitzem la construcció de la funció potencial. En el sisè, mostrem com fem per moure els mòduls durant la reconfiguració a la forma canònica. I en el setè, el procés que fem per preparar els mòduls per generar una forma nova.

Cadascun d'aquests sis capítols està estructurat en set apartats. En el primer apartat expliquem l'objectiu de la fase, allò que volem aconseguir. En el segon, l'estratègia, com ho fem per aconseguir el nostre objectiu i quin és el procés que hem seguit. En el tercer apartat fem una descripció de les regles utilitzades. En el quart, comentem els problemes que hem trobat i quina solució hem adoptat. En el cinquè apartat expliquem quines modificacions s'han anat fent a mida que avançàvem. En el sisè, exposem les alternatives que havíem plantejat i que finalment no s'han fet servir. En l'últim apartat, el setè, expliquem alguns dels jocs de proves utilitzats per verificar el funcionament de la fase. Els capítols dos i tres, pel fet que les regles van ser implementades externament a aquest projecte, hem substituït els apartats de regles i alternatives per un apartat d'adaptació on expliquem quines modificacions hem fet per adaptar tot el conjunt.

En el vuitè capítol fem una descripció del simulador, les seves funcionalitats i les modificacions realitzades. En el novè capítol s'explica la gestió del projecte i el seu pressupost. En el desè s'exposen les conclusions sobre el resultat del projecte, les dificultats que hem trobat i propostes per futures ampliacions.

La memòria inclou la bibliografia utilitzada per dur a terme aquest treball. També s'adjunta un annex amb totes les regles detallades dels capítols 4, 5, 6 i 7.



## 2 Cerca del màster

### 2.1 Objectiu

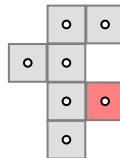
Es tracta de trobar, d'entre tots els mòduls que formen la configuració del robot, quin d'ells farà la funció de màster.

El màster és el mòdul que determina l'inici i final de la reconfiguració. És el que inicia l'enviament de les dades i missatges necessaris per les fases posteriors, i també l'últim que rep la informació del conjunt del robot.

El màster és l'únic mòdul que manté la seva posició durant tot el procés de reconfiguració. La resta de mòduls es mouran en les següents fases buscant enganxar-se a la tirallonga que es generarà a partir d'aquest mòdul.

### 2.2 Estratègia

El màster és el mòdul que hi ha dels de més a la dreta, el de més avall (exemple a la Figura 2.1).



**Figura 2.1: Exemple de configuració.** Es mostra quin dels mòduls és el màster (mòdul vermell).

El màster ha de ser un dels mòduls que es troben en la frontera externa del robot. Amb aquesta decisió ens assegurem que no existeix cap mòdul ni a la dreta ni a sota del màster, i és a la dreta on s'aniran acumulant els mòduls que conformen la configuració canònica.

El mòdul màster es pot assignar manualment donant-li l'estat característic de màster [\*\*\*\*M] o mitjançant les regles. Tot mòdul sap si és o no màster per mitjà de l'últim caràcter de l'estat, la [M] (*Master*) indica que és el màster del sistema.

Fent servir les regles, la configuració inicial parteix de la base que tots els mòduls es troben enganxats als seus veïns i en estat de indefinició [UNDEF].

A l'iniciar l'aplicació de les regles es determinen quins són els candidats a màster i quins no (la selecció de candidats a forat es fa en el mateix instant). Un candidat a màster és aquell mòdul que no té cap veí ni al Sud, ni al Est.

Els mòduls que han estat seleccionats com a candidats passen a l'estat de [iniM0], on ini es refereix a la fase inicial (detecció de màster i forats) i M0 determina que aquell mòdul és candidat a màster.

Aquests mòduls són els encarregats de propagar un missatge que recorre tota la frontera del robot on es troba el candidat (exterior o d'un forat), tot seguint la regla de la mà dreta, fins que retorna a ell mateix amb la informació de si passa a ser finalment el màster del sistema, o pel contrari és un mòdul normal. El missatge per defecte és afirmatiu i deixa de ser-ho si passa per un mòdul que es troba més a la dreta o que, trobant-se a la mateixa vertical, està per sota del candidat.

La circulació dels missatges es pot identificar per l'estat del mòdul. Aquell mòdul pel que passa un missatge positiu té l'estat de [iniCM] (*Candidate Master*), en canvi, per aquell mòdul que passa el missatge negatiu té l'estat de [iniCS] (*Candidate Slave*). Aquests estats són únicament per l'ajuda visual de la circulació dels missatges i tenen una durada limitada a un únic "torn".

Un cop arriba el missatge al candidat, si és positiu (CYES1) aquell mòdul passa a l'estat [bst\_M], on bst és la nomenclatura de la fase següent i M és el que l'identifica com a màster del sistema; en canvi si el missatge és negatiu (CNO1\_) el candidat deixa de ser-ho i passa a la fase següent com la resta de mòduls, amb estat [bst\_ \_].

Cada missatge surt del candidat amb la informació de la seva posició relativa que es va actualitzant a mida que el missatge avança al llarg de la frontera. Així, quan li retorna, un candidat sap que aquell missatge li correspon a ell perquè va acompanyat de les seves coordenades relatives (0,0).

## **2.3 Adaptació**

En un principi tota aquesta fase es feia de forma manual indicant quin era el màster modificant el seu estat.

Les regles que permeten la detecció de màster van ser implementades externament a aquest projecte i, posteriorment, incorporades a aquest quan ja estava completada la fase principal. Per tant, ha estat necessari adaptar les dues parts per aconseguir que no hi hagués conflictes.

L'adaptació s'ha realitzat en diverses parts:

· Els noms dels estats han estat modificats per tal que tots els relacionats amb la cerca del màster iniciessin amb [ini], indicant que totes aquestes regles pertanyen a la fase inicial. D'aquesta manera s'identifiquen els mòduls de la forma següent:

[iniM0]	Candidat a màster ( <i>Master</i> )
[iniCM]	Missatge de candidat a màster YES ( <i>Candidate Master</i> )
[iniCS]	Missatge de candidat a màster NO ( <i>Candidate Slave</i> )

· Les prioritats s'han unificat de manera que es poden aplicar diverses regles paral·lelament, això és necessari per fer que els mòduls puguin comunicar independentment els missatges que busquen al màster i els que transporten altres fases del procés de reconfiguració.

· Els comptadors que es fan servir per conèixer l'estat en que queda un mòdul s'han eliminat perquè la informació que s'extreia d'aquests comptadors era la mateixa que oferia el propi estat.

· Respecte a les regles, per una part s'han eliminat estats intermedis que identificaven els mòduls que transportaven els missatges. Per altra banda s'han afegit regles que realitzen la transició de la fase de la cerca del màster a la següent, la de la construcció de l'*scan tree*. També s'han fet noves regles per identificar els mòduls que transporten els missatges dels candidats a màster. Tot i que són prescindibles, ajuden a la visualització del procés al simulador.

· Els canals de comunicació s'han substituït per uns de nous per diferenciar-los dels que fa servir la fase de construcció de la funció potencial. Aquest canvi serveix per a què cada tipus de missatge tingui el seu canal de comunicació, tot i que en aquest cas no produiria cap conflicte fer servir els mateixos canals.

## 2.4 Problemes

Destaquem a continuació les qüestions principals que han sorgit en implementar aquesta fase i la solució adoptada en cada cas.

**2.4.1 Prioritats:** les prioritats són una forma simple d'estalviar precondicions. La part positiva és que simplifiquen molt les regles, però per contra impedeixen que es pugui executar més d'una regla alhora si les seves prioritats són diferents.

*Solució adoptada: en aquest cas ens ha semblat adequat que regles de fases diferents es puguin executar alhora, per tant hem de prescindir de les prioritats i convertir-les en precondicions.*

**2.4.2 Estats intermedis:** un mòdul només pot tenir un únic estat i l'aplicació de les regles ve determinada principalment per l'estat del mòdul. En viatjar simultàniament diversos missatges, es barregen canvis d'estats de dues fases diferents i, en trobar-se dos d'ells, només un sobreviu.

*Solució adoptada: la fase de cerca del màster envia els missatges sense modificar l'estat del mòdul, això fa que els missatges es puguin creuar sense crear conflictes, però no puguem identificar de forma visual per on es mouen aquests missatges (vegeu el problema de l'Apartat 2.4.4).*

**2.4.3 Canals de comunicació:** realment aquí no existeix cap problema (que sí existeix en la fase de reconfiguració inversa). Tanmateix, ens ha semblat millor que cada tipus de missatge tingui el seu propi canal, a banda de mantenir la coherència ja que en una altra fase el canvi s'ha de fer de forma obligada. Aquí no es pot donar el conflicte degut a què els missatges de màster i de valor potencial (d'anada) circulen per fronteres del robot que són diferents que per les que circulen els missatges de forat i de valor potencial (de tornada), per tant, es poden creuar però no anar alhora en la mateixa direcció pel mateix canal. Per altra banda, entre els que circulen en un mateix sentit tampoc no es pot produir conflicte ja que el missatge de valor potencial (d'anada) no pot iniciar-se fins que el de màster no s'ha resolt. I els de valor potencial (de tornada) i forat tampoc no poden generar conflictes degut a què les regles que transmeten el valor potencial es mantenen a l'espera en trobar-se amb un candidat encara no resolt.

*Solució adoptada: modificacions en el simulador per ampliar els canals de comunicació.*

**2.4.4 Visualització dels missatges:** degut a què la comunicació dels missatges dels candidats no modifiquen l'estat dels mòduls, aquests no poden tenir assignat cap color que els identifiqui al simulador.

*Solució adoptada: s'han dissenyat i implementat regles específiques que només afecten els mòduls d'un estat concret [bpf\_ \_] que reben un missatge d'un candidat, però no reben cap missatge de la fase que s'encarrega de la funció potencial. Això produeix un canvi d'estat momentani (la seva durada és d'un sol "torn") per tal de visualitzar per on ha passat un missatge de candidat. Només afecta els mòduls que es troben en aquell estat concret i no a la resta de fases. Fer aquest canvi només si no es rep un missatge de valor potencial assegura que durant aquest "torn" no es produeixen canvis d'estat al mòdul afectat i per tant pot retornar a l'estat [bpf\_ \_] sense produir conflictes. El cas de solapament de missatges en un estat intermedi es tracta a l'apartat següent (problema de l'Apartat 2.4.5).*

**2.4.5 Solapament dels colors:** les regles de la fase de construcció de la funció potencial es basen en l'estat dels mòduls, per tant pot donar-se el cas que el missatge de valor potencial vagi a parar a un mòdul que ha canviat de l'estat [bpf\_ \_] a l'estat intermedi de visualització de missatge de forat o màster. Aquesta és una situació que no estava contemplada, per tant el mòdul afectat no tenia cap regla per aplicar i el missatge es perdia.

*Solució simple: eliminar les regles de colors. Si s'eliminen les regles que permeten visualitzar el recorregut dels missatges al simulador, mai no es donaria aquest cas, ja que els mòduls mantindrien el seu estat original, però això provocaria la pèrdua visual de per on circulen els missatges dels candidats.*

*Solució adoptada: introduir regles noves que contemplen aquests casos. Es generen noves regles per tal que els mòduls amb estats intermedis (de visualització) siguin capaços de continuar enviant el missatge del valor potencial i canviar al nou estat que els pertoca en funció de la regla aplicada (vegeu les regles a l'Apartat 5.3.8).*

Com es pot observar, la major part de dificultats trobades i resoltes en aquesta fase estan relacionades amb l'objectiu de visualització del pas de missatges per part del simulador, més que no pas pel pas mateix dels missatges entre els mòduls del robot.

## 2.5 Modificacions

En un principi la cerca del màster es feia de forma simulada, identificant el màster de forma manual. S'inicialitzava el mòdul que seria el màster amb l'estat [bst\_M] i la resta amb l'estat [bst\_ \_], llavors les regles començaven a partir de la fase de "construcció de l'scan tree" amb el màster (i els forats) ja marcats.

Posteriorment, es van incorporar a aquest projecte les regles que feien aquest procés de forma automàtica provenint d'un altre projecte. Aleshores, es van adaptar a la resta de regles. Inicialment es feien servir prioritats, i es van modificar perquè fossin superiors a les que tenien les fases posteriors, ja que aquestes regles s'havien d'aplicar en primer lloc.

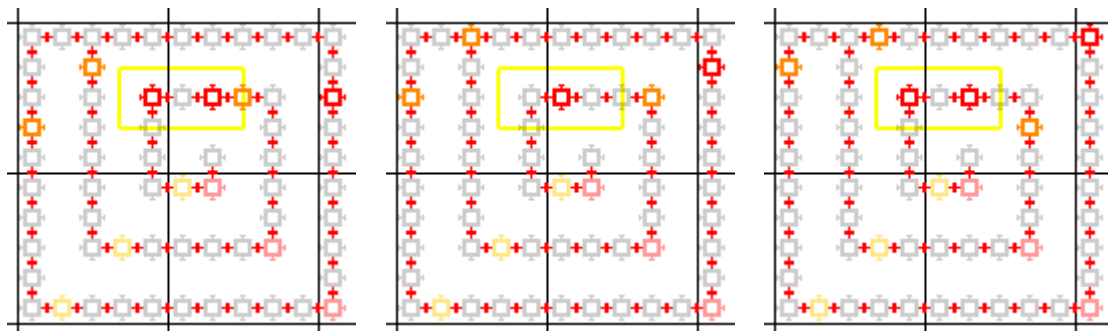
Degut a la necessitat que aquesta fase funcionés en paral·lel amb la resta, finalment es van unificar les prioritats de totes les fases implicades.

Per altra banda, en un principi es van canviar els comptadors a uns altres lliures ja que utilitzava els mateixos que en fases posteriors. Finalment, com que feien la mateixa funció que l'estat del mòdul, es van deixar de fer servir.

## 2.6 Models de prova

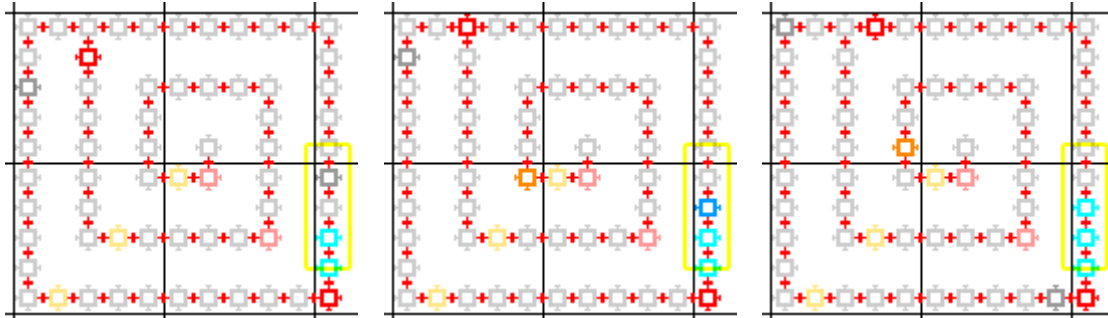
S'han fet experiments per comprovar tots els conflictes possibles en trobar-se missatges de fases diferents. També per comprovar el recorregut dels colors al simulador, el solapament dels colors i les fases. Aquests models contempnen:

**2.6.1 Solapament de missatges de la mateixa fase:** es produeix quan un mateix mòdul rep missatges de dues direccions. Aquest mòdul es capaç de gestionar-los de forma que cada missatge continua pel camí que pertoca. Aquesta gestió es realitza correctament perquè les regles treballen amb la mateixa prioritat i amb precondicions independents, això comporta que el mòdul gestiona cada missatge per separat. Aquests missatges no poden coincidir en la mateixa direcció degut a què tots els missatges inicien el recorregut d'una mateixa frontera alhora, per tant sempre hi ha la mateixa distància entre missatges que entre candidats. I tampoc no pot passar entre missatges que circulen un per la frontera interior de la configuració i l'altre per l'exterior, ja que aleshores les direccions que segueixen no coincideixen (tal com s'il·lustra a la Figura 2.2).



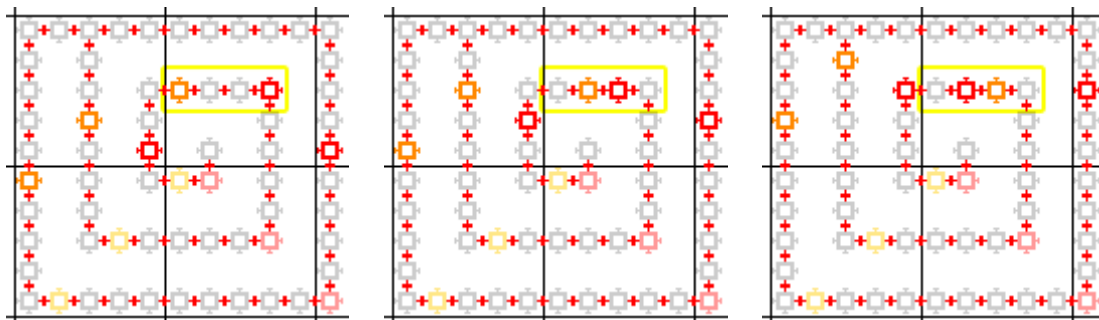
**Figura 2.2: Solapament de dos missatges de candidat de màster.** Es mostren tres “torns” consecutius centrats en els tres mòduls afectats. En el primer torn el mòdul vermell de l’esquerra transporta el missatge cap a la dreta, i el de la dreta cap a l’esquerra. En el segon torn tots dos missatges es solapen al mateix mòdul. I en el tercer els missatges continuen el seu camí, el que abans estava a l’esquerra ara es troba a la dreta, i el que estava a la dreta ara està a l’esquerra.

**2.6.2 Solapament de missatges de fases diferents:** a l'igual que els de la mateixa fase, com que apliquen les regles de forma independent, no interfereixen uns missatges amb altres. Però en aquest cas es pot donar que el mòdul al que ha d'enviar el valor potencial es trobi en l'estat de canvi de color degut a un missatge de candidat a forat o màster. Si es dona aquesta situació, el mòdul passa a estat d'espera fins que la situació es resol al torn següent (tal com s'il·lustra a la Figura 2.3).



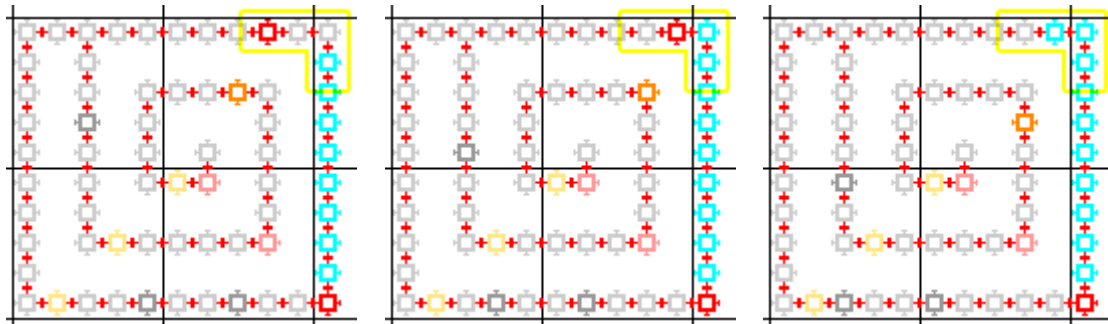
**Figura 2.3: Solapament de dos missatges de fases diferents.** Es mostren tres "torns" consecutius centrats en els tres mòduls afectats. En el primer torn el mòdul de dalt envia un missatge de candidat cap a baix, i el de baix envia un missatge de valor potencial cap a munt. En el segon torn tots dos missatges es solapen al mateix mòdul i mentre que el missatge de candidat continua la seva marxa, el de valor potencial es manté a l'espera (mòdul blau fosc). En el tercer torn, el missatge de valor potencial reprèn la marxa (mòdul blau cel).

**2.6.3 Creuament de missatges de la mateixa fase:** això no genera cap incidència. Els mòduls que abans enviaven el missatge ara el reben i el continuen enviant per allà on toca (tal com s'il·lustra a la Figura 2.4).



**Figura 2.4: Creuament de dos missatges de candidat diferents.** Es mostren tres "torns" consecutius centrats en els quatre mòduls afectats. En el primer torn el mòdul taronja de l'esquerra transporta el missatge cap a la dreta, i el mòdul vermell de la dreta el transporta cap a l'esquerra. En el segon torn cada mòdul envia el seu missatge a on es troba l'altre. I en el tercer els missatges s'han creuat i continuen el seu camí, el que abans estava a l'esquerra ara es troba a la dreta, i el que estava a la dreta ara està a l'esquerra.

**2.6.4 Creuament de missatges de fases diferents:** igual que els de la mateixa fase, però com que les regles de la construcció de la funció potencial varien l'estat del mòdul, aquest modificarà el seu aspecte visual deixant de veure's la circulació dels missatges dels candidats fins que no passin per una altra zona en la qual els mòduls encara estan sense el valor potencial assignat (tal com s'il·lustra a la Figura 2.5). És també en aquest cas quan s'apliquen les observacions fetes arran del problema del solapament de colors (problema de l'Apartat 2.4.5).



**Figura 2.5: Creuament de dos missatges de fases diferents.** Es mostren tres “torns” consecutius centrats en els quatre mòduls afectats. En el primer torn el mòdul vermell envia un missatge de candidat cap a la dreta, i el blau envia un missatge de valor potencial cap a munt. En el segon torn cada mòdul envia el seu missatge a on es troba l'altre. En el tercer torn els missatges s'han creuat i continuen el seu camí, el que abans estava a l'esquerra ara es troba a la dreta (no s'identifica visualment), i el que estava a la dreta ara està a l'esquerra (el blau).



## 3 Detecció de forats

### 3.1 Objectiu

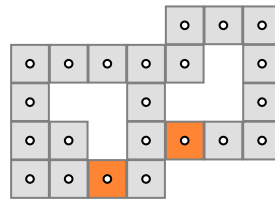
Es tracta de detectar si hi ha forats dintre de la configuració del robot, i quin serà el mòdul responsable de cada forat, aquest mòduls els anomenem líders de forat.

Els líders de forats són els encarregats de desfer aquell forat del que formen part, d'aquesta manera aconseguim que l'arbre que volem generar no contingui cap cicle.

A diferència del màster, que és únic, líders de forats hi ha tants com forats tingui la configuració.

### 3.2 Estratègia

El líder de forat és el mòdul que hi ha a sota del buit dels de més avall, el de més a l'esquerra (exemple a la Figura 3.1).



**Figura 3.1: Exemple de configuració amb dos forats.** Es mostren quins mòduls són els líders de forat (mòduls en taronja).

El líder de forat ha de ser un dels mòduls que conformen el forat, ja que serà aquest mòdul el que trenqui una de les seves connexions per tallar el cicle que produeix un forat en el graf d'adjacències del robot.

El mòdul responsable de cada forat es pot assignar manualment donant-li l'estat característic de líder de forat [\*\*\*\*H] o mitjançant les regles. Tot mòdul sap si és o no líder de forat per mitjà de l'últim caràcter de l'estat, la [H] (*Hole*) indica que és un líder de forat.

La configuració inicial parteix de la base que tots els mòduls es troben enganxats als seus veïns i en estat de indefinició [UNDEF].

A l'iniciar l'aplicació de les regles, es determina quins són els candidats a líder de forat i quins no (a l'igual que succeïa amb el màster). Un candidat a líder de forat és aquell mòdul que no té cap veí al Nord, però sí que en té a l'Est, a l'Oest, i al Nord-Oest.

Els mòduls que han estat seleccionats com a candidats passen a l'estat de [iniH0], on ini representa la fase inicial (detecció de màster i forats) i H0 determina que aquell mòdul és un candidat a líder de forat.

Aquests mòduls són els encarregats de propagar un missatge que recorre la frontera del robot on es troba cada candidat. Això vol dir que, en cas de ser un forat, estaria passant per tots els mòduls que envolten aquell forat. Al retornar el missatge al mòdul candidat, aquest obté la informació de si aquest recorregut forma part o no d'un forat, i si aquest mòdul passa a ser un líder de forat o un mòdul normal. Aquest missatge per defecte és afirmatiu i deixa de ser-ho si passa per un mòdul que es troba per sota o que, trobant-se a la mateixa alçada, està més a l'esquerra que el candidat.

La circulació dels missatges es pot identificar per l'estat del mòdul. Aquell mòdul pel que passa un missatge positiu té l'estat de [iniCH] (*Candidate Hole*), en canvi, aquell mòdul que passa un missatge negatiu té l'estat de [iniCL] (*Candidate Line*). Aquests estats són usats únicament per l'ajuda visual de la circulació dels missatges i tenen una durada limitada a un únic "torn".

Un cop arriba el missatge al candidat, si és positiu (CYES2) aquell mòdul passa a l'estat [bst\_H], on bst és la nomenclatura de la fase següent i H és el que l'identifica com a líder de forat; en canvi si el missatge és negatiu (CNO2\_) el candidat deixa de ser-ho i passa a la fase següent com la resta de mòduls, amb estat [bst\_ \_].

Un candidat sap que un missatge li correspon perquè va acompanyat de les seves coordenades, com en el cas dels missatges dels candidats a màster.

### **3.3 Adaptació**

L'adaptació d'aquesta fase segueix les mateixes pautes que l'adaptació de la cerca del màster. En un principi tota aquesta fase es feia de forma manual indicant quin era el líder de forat modificant el seu estat.

També com en el cas anterior, les regles per a la detecció dels forats es van incorporar, provenint d'un altre projecte, un cop completada la fase principal. Per tant, era necessari adaptar les dues parts per aconseguir que no hi hagués conflictes.

L'adaptació és molt similar a la descrita a l'Apartat 2.3, per això es comenta breument a continuació, indicant les diferències respecte a l'adaptació feta amb la cerca del màster:

· Els **noms dels estats** han estat modificats per tal que tots els relacionats amb la detecció dels forats iniciessin amb [ini], indicant que totes aquestes regles pertanyen a la fase inicial. D'aquesta manera s'identifiquen els mòduls de la forma següent:

[iniH0]	Candidat a líder de forat ( <i>Hole</i> )
[iniCH]	Missatge de candidat a líder de forat YES ( <i>Candidate Hole</i> )
[iniCL]	Missatge de candidat a líder de forat NO ( <i>Candidate Line</i> )

· Les **prioritats** també s'han unificat de manera que es poden aplicar diverses regles paral·lelament.

· Com amb el màster, els **comptadors** que es fan servir per conèixer en quin estat queda un mòdul s'han eliminat.

· Respecte a les **regles**, també s'han eliminat els estats intermedis, s'han afegit les regles que realitzen la transició de la fase de detecció de forats a la següent, i s'han fet les noves regles per identificar els mòduls que transporten els missatges dels candidats a líder de forat.

· Els **canals de comunicació** s'han substituït pels mateixos que fan servir els candidats a màster.

## 3.4 Problemes

De la mateixa forma que l'adaptació és molt similar a la realitzada en la fase de cerca del màster, els problemes ocasionats també són equivalents als descrits a l'Apartat 2.4. Per això aquí s'enumeren i s'expliquen les diferències (si n'hi ha) respecte a les solucions adoptades en aquest casos.

**3.4.1 Prioritats:** el problema és equivalent al del màster (*vegeu el problema 2.4.1*).

*Solució adoptada: igual que amb les regles de la cerca del màster, hem transformat les prioritats en precondicions.*

**3.4.2 Estats intermedis:** el problema és equivalent al del màster (*vegeu el problema 2.4.2*).

*Solució adoptada: en aquest cas, són els missatges de candidat de forat els que s'envien sense modificar l'estat del mòdul. De la mateixa manera que passava amb el màster, també deixem de poder identificar visualment per on es mouen aquests missatges (vegeu el problema de l'Apartat 3.4.4).*

**3.4.3 Canals de comunicació:** el problema és equivalent al del màster (vegeu el problema 2.4.3).

*Solució adoptada: es fa servir la mateixa modificació en el simulador ja que utilitza els mateixos canals emprats pels missatges de candidat a màster.*

**3.4.4 Visualització dels missatges:** el problema és equivalent al del màster (vegeu el problema 2.4.4).

*Solució adoptada: s'han duplicat les regles fetes servir en el cas (2.4.4) però adaptant-les als missatges concrets que transporten els candidats a líder de forat. De forma anàloga al màster, el solapament de missatges en un estat intermedi té un tractament específic (vegeu l'Apartat 3.4.5).*

**3.4.5 Solapament dels colors:** el problema és equivalent al del màster (vegeu el problema 2.4.5).

*Solució simple: a l'igual que amb el màster, eliminant les regles de colors, aquest problema mai no es donaria.*

*Solució adoptada: en aquest cas es modifiquen les regles utilitzades per solucionar el problema que es produïa amb el màster (Apartat 2.4.5) per incloure a les seves precondicions els casos dels candidats a líder de forat (vegeu les regles a l'Apartat 5.3.8).*

Com es pot observar, els problemes derivats per la detecció dels líders de forat són els mateixos que els que ocasionava la fase de cerca del màster. Això es degut a què, tot i que cada fase té les seves pròpies regles, totes dues segueixen processos molt similars.

## **3.5 Modificacions**

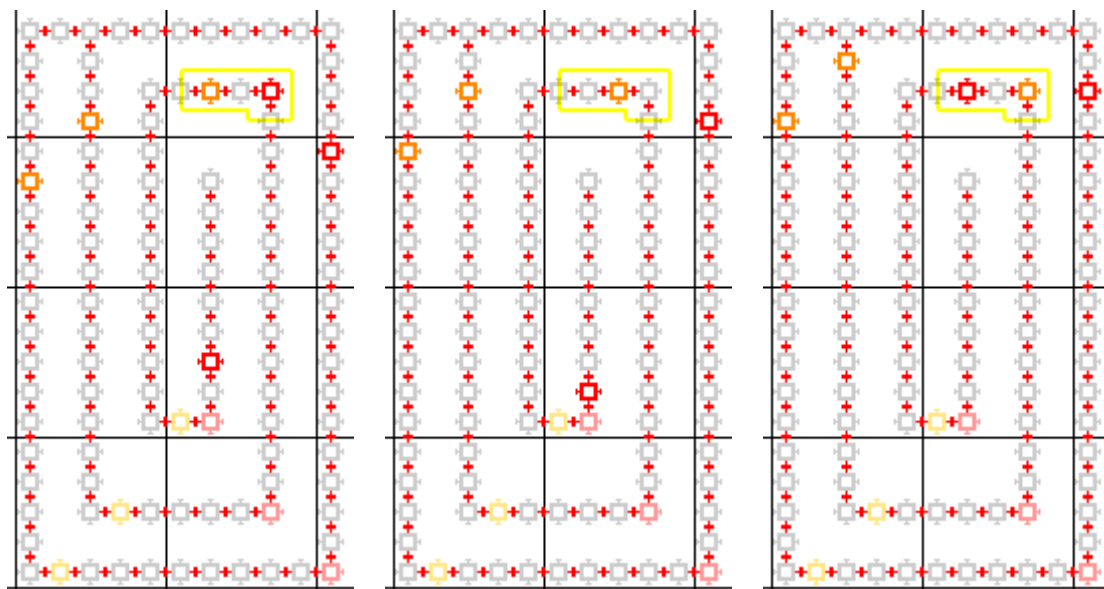
Tal com passava amb el màster, en un principi tots els líders de forat es marcaven de forma manual. S'iniciaven tots els mòduls que serien líders de forat amb l'estat [bst\_H] i la resta amb l'estat [bst\_ \_], llavors les regles començaven a partir de la fase de "construcció de l'scan tree" amb els líders de forats (i el màster) ja detectats.

Juntament amb les regles de màster, es van incorporar les regles per detectar forats. A continuació es va seguir el mateix procés d'adaptació: modificació de les prioritats per, finalment, unificar-les; i modificació dels comptadors per, posteriorment, deixar de fer-los servir.

## 3.6 Models de prova

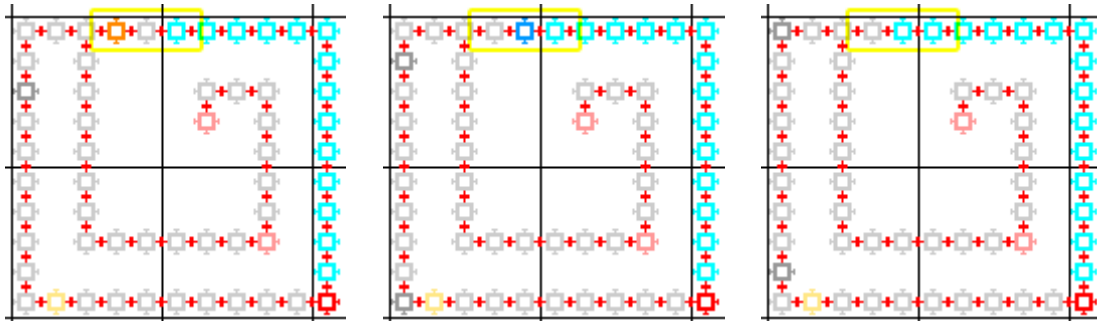
En aquest cas s'han fet experiments per comprovar els possibles conflictes en trobar-se missatges de tipus diferents (màster i forat). I també per verificar que el solapament i creuament de missatges segueix sense produir conflictes en el cas de missatges de candidats a líders de forat:

**3.6.1 Solapament de missatges de la mateixa fase:** es produeix quan un mateix mòdul rep missatges de dues direccions. El sistema és similar al cas (Apartat 2.6.1), però en aquesta ocasió els missatges són de dos tipus diferents, un és de màster i l'altre de forat. El mòdul que rep els missatges es capaç de gestionar-los de forma que cada missatge continua pel camí que pertoca, independentment de si són de tipus diferents (tal com s'il·lustra a la Figura 3.2). Això és degut a què el procés que segueixen tots dos missatges és el mateix.



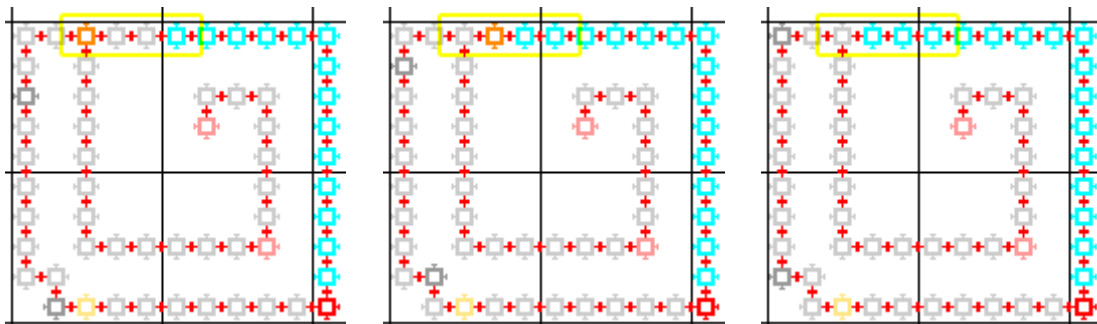
**Figura 3.2: Solapament de missatges de màster i forat.** Es mostren tres "torns" consecutius centrats en els tres mòduls afectats. En el primer torn el mòdul taronja transporta el missatge de forat cap a la dreta, i el mòdul vermell el missatge de màster de la dreta cap a l'esquerra. En el segon torn tots dos missatges es solapen al mateix mòdul (tot i ser taronja, conté els dos missatges). I en el tercer torn els missatges continuen el seu camí, el de forat ara es troba a la dreta, i el de màster continua per l'esquerra.

**3.6.2 Solapament de missatges de fases diferents:** tal i com passa amb el màster (Apartat 2.6.2) el missatge de candidat a forat continua, mentre que el de valor potencial es manté un torn a l'espera (tal com s'il·lustra a la Figura 3.3).



**Figura 3.3: Solapament de dos missatges de fases diferents.** Es mostren tres “torns” consecutius centrats en els tres mòduls afectats. En el primer torn el mòdul taronja envia un missatge de candidat cap a la dreta, i el blau envia un missatge de valor potencial cap a l'esquerra. En el segon torn tots dos missatges es solapen al mateix mòdul i el missatge de valor potencial es manté a l'espera (mòdul blau fosc). En el tercer torn, el missatge de valor potencial reprèn la marxa (mòdul blau cel).

**3.6.3 Creuament de missatges:** els resultats de les proves són equivalents als realitzats en els casos dels missatges de màster (apartats 2.6.3 i 2.6.4). A la Figura 3.4 es pot observar una de les proves realitzades amb un missatge de forat implicat.



**Figura 3.4: Creuament de dos missatges de fases diferents.** Es mostren tres “torns” consecutius centrats en els quatre mòduls afectats. En el primer torn el mòdul taronja envia un missatge de candidat cap a la dreta, i el blau un de valor potencial cap a l'esquerra. En el segon torn cada mòdul envia el seu missatge a on es troba l'altre. En el tercer torn els missatges s'han creuat i continuen el seu camí.

## 4 Construcció de l'*scan tree*

### 4.1 Objectiu

Es tracta de transformar la configuració del robot en un graf connex acíclic, de tal manera que l'arbre lògic coincideixi amb les connexions entre mòduls.

Les fulles de l'arbre han de trobar-se a la frontera de la configuració, ja que posteriorment seran els mòduls fulla els que s'activaran i produiran la reconfiguració.

### 4.2 Estratègia

A l'inici el robot està completament connectat, per això les regles han de decidir quins mòduls s'han de desconnectar.

El graf resultant de les regles és una estructura en la qual tots els mòduls estan units verticalment.

Si la configuració no té forats, cada columna es manté unida a les veïnes a través de la parella de mòduls de més amunt, (el graf resultant és trivialment connectat i acíclic). O el que és el mateix, si un mòdul té veïns al seu Nord i Nord-Est aquest pot desconnectar-se del seu veí Est perquè les seves columnes ja estaran unides per aquesta (Nord, Nord-Est) o altra parella existent més amunt que no compleixi aquesta regla, i per tant, no es desconnecti. De la mateixa manera es comporta amb veïns Nord, Nord-Oest, desconnectant-se, en aquest cas, del seu Oest.

Si la configuració té forats, el líder de forat es desconnecta per la seva esquerra (Oest) per no crear cicles. Aquests líders de forat s'identifiquen prèviament en la fase de detecció de forats (vegeu el Capítol 3).

La resta de mòduls que no són forats i no tenen cap veí al Nord només han de realitzar el canvi d'estat. Aquests es mantenen connectats amb els seus veïns, fent que siguin els responsables de connectar dues columnes entre elles.

Per tant, el procés a seguir és el següent (vegeu la Figura 4.1 pe un exemple):

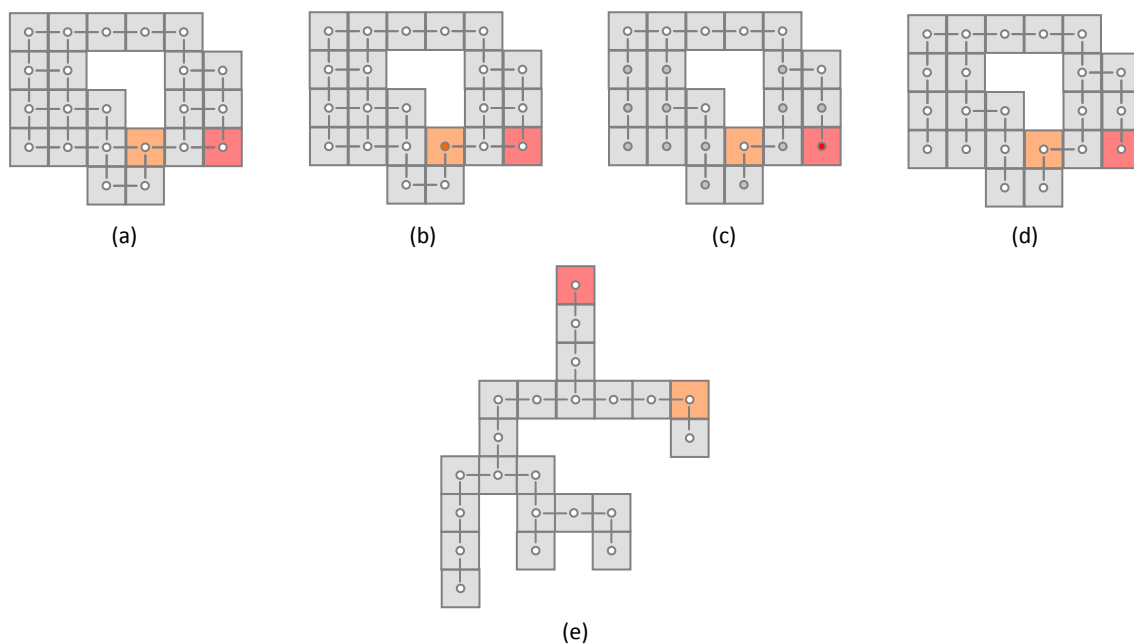
Inicialment tots els mòduls estan connectats entre ells (Figura 4.1.a).

Tots els mòduls identificats com a líders de forat es desconnecten del seu Oest, trencant així el forat del que són responsables (Figura 4.1.b).

Aquells mòduls que tenen el seu Nord i Nord-Est ocupat es desconnecten del seu Est; i els que tenen el seu Nord i Nord-Oest ocupat desconnecten el seu Oest (Figura 4.1.c).

La resta de mòduls es mantenen connectats i tots realitzen el canvi d'estat per passar a la fase següent (Figura 4.1.d).

La configuració resultant són tires de mòduls connectades verticalment, unides per la parella de mòduls de més amunt i amb tots els cicles trencats pels líders de forat. Aquesta estructura és un arbre, el nostre *scan tree*. Es pot veure la forma d'arbre més fàcilment a la Figura 4.1.e.



**Figura 4.1: Procés de construcció de l'arbre.** Es mostra el procés de desconexió dels mòduls, i l'arbre resultant. (a) Estat inicial. (b) Desconexió del líder de forat (mòdul taronja). (c) Separació de les branques. (d) Estructura resultant. (e) Arbre expandit equivalent.



## 4.3 Regles

Per realitzar aquesta fase s'han fet servir 4 regles. Una per tractar els líders de forat, dues per separar les branques, i l'última per passar a la fase següent.

Les regles d'aquesta fase s'identifiquen per les sigles [bst] (*Build Scan Tree*). Aquests tres caràcters a l'estat d'un mòdul serveixen al simulador per identificar la fase en la qual es troba. L'últim caràcter, en canvi, serveix per diferenciar el tipus de mòdul que és. D'aquesta manera, identifiquem els mòduls de la forma següent:

[bst_M]	Màster en la fase build scan tree
[bst_H]	Líder de forat en la fase build scan tree
[bst_ ]	Resta de mòduls en la fase build scan tree

Per veure la descripció detallada de les regles, vegeu l'Annex A.

## 4.4 Problemes

No s'ha presentat cap problema significatiu durant la realització d'aquesta fase.

## 4.5 Modificacions

En un principi aquesta fase era la inicial, ja que el màster i els líders de forats s'introduïen manualment i la resta de mòduls ja tenien l'estat de bst. Això provocava que en iniciar les regles tots els mòduls començaven la fase de construcció de *l'scan tree* alhora.

En una primera versió els mòduls iniciaven tots desconnectats i es connectaven mitjançant l'aplicació de les regles. Més tard es va optar per l'opció contrària, que també era més coherent: que el robot estigués totalment connectat i fossin les regles les que decidissin per on s'havien de desconnectar els mòduls per donar forma a l'arbre que representaria la configuració del robot.

Posteriorment, un cop obtingudes les regles que feien la detecció de màster i forats de forma automàtica, es van adaptar les regles. Aquesta adaptació només consistia en la modificació d'algunes precondicions per equiparar totes les regles a una mateixa prioritat. Es va aprofitar aquesta unificació de prioritats per fer una regla comuna de canvi de fase. En lloc que cada regla realitzés aquest canvi, és la regla que afecta tots els mòduls de la fase l'encarregada de fer el pas a la següent simultàniament amb la resta de regles de desconnexió.

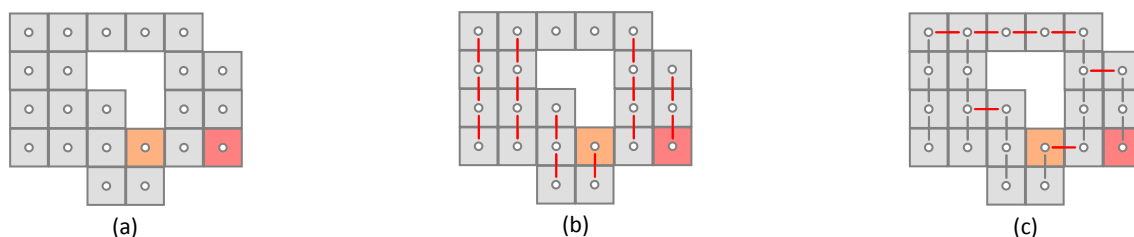
## 4.6 Alternatives

Una alternativa per aquesta fase és la que es va implementar en un principi. Suposant que els mòduls inicialment estan desconnectats, mitjançant l'aplicació de les regles, es tracta d'anar connectant-los fins a aconseguir el graf objectiu (Figura 4.2.a).

La idea és la mateixa que la que hem aplicat però de forma inversa. Primerament, tots els mòduls s'uneixen verticalment (Figura 4.2.b).

Si la configuració no té forats, cada columna s'uneix a la columna veïna a través de la parella de mòduls de més amunt. En canvi, si la configuració té forats, el líder de forat (que també formarà part d'una de les parelles de més amunt) evita fer la connexió amb la columna de la seva esquerra evitant així de formar un cicle (Figura 4.2.c).

L'arbre resultant acaba sent el mateix.



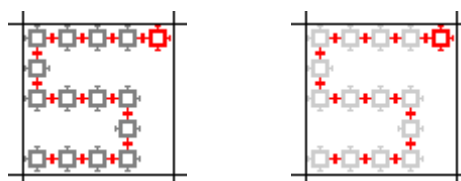
**Figura 4.2: Procés alternatiu de construcció de l'arbre.** Es mostra el procés de connexió dels mòduls. (a) Estat inicial. (b) Connexió de les branques verticals. (c) Unió de les branques (llevat de les branques on es troben els líders de forat).

## 4.7 Models de prova

Els experiments han consistit en múltiples formes per verificar que les desconnexions es realitzen de forma correcta. Inclouen formes simples, formes complexes, sense forats i amb múltiples forats.

Alguns exemples són els que tractem a continuació:

**4.7.1 Forma de cuc:** no necessita cap desconnexió, simplement passen a la fase següent (Figura 4.3).



**Figura 4.3: Configuració en forma de cuc.**

**4.7.2 Forma densa:** es pot observar el resultat de la ramificació en columnes (Figura 4.4).

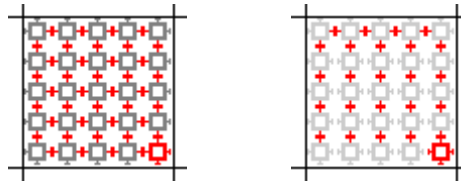


Figura 4.4: Configuració densa.

**4.7.3 Forma semi-densa:** es pot observar el resultat de la ramificació en columnes tant de branques principals com de subbranques (Figura 4.5).

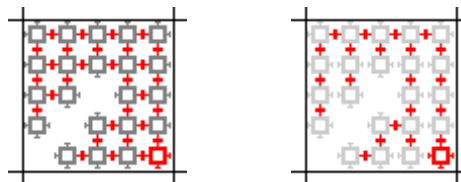


Figura 4.5: Configuració semi-densa.

**4.7.4 Forat simple:** el líder de forat trenca el cicle ocasionat pel forat, separant-se de la columna Oest adjacent (Figura 4.6).

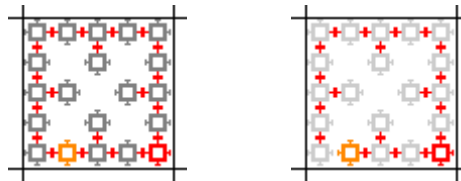


Figura 4.6: Configuració amb un forat.

**4.7.5 Forats múltiples:** els dos líders de forat trenquen el cicle del que són responsables amb la seva desconexió (Figura 4.7).

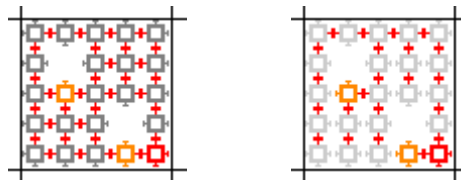


Figura 4.7: Configuració amb dos forats.



## 5 Construcció de la funció potencial

### 5.1 Objectiu

Un cop tenim l'*scan tree* de la configuració del robot i s'ha determinat quin és el mòdul màster (que fa la funció d'arrel de l'arbre), es tracta d'assignar a cada mòdul un valor de funció potencial. Aquest valor indica la distància d'un mòdul a l'arrel de l'arbre en fer un recorregut en profunditat.

Al finalitzar la construcció de la funció potencial cada mòdul sap, a més del seu valor potencial, a quina de les diferents branques que conformen l'arbre pertany, i qui és el seu mòdul pare. Per branca entenem cada camí de l'arbre format per mòduls de grau 2 o 1.

Aquestes dades serveixen per determinar els moviments dels mòduls en la fase següent.

### 5.2 Estratègia

A l'inici d'aquesta fase l'arbre està construït i el màster assignat.

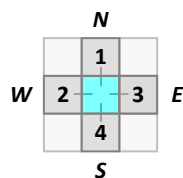
La construcció de la funció potencial es fa estenent el seu valor, començant pel màster, seguint per les connexions de l'arbre. La propagació d'aquests missatges segueix un recorregut en profunditat amb prioritat antihorària iniciat des del màster. Aquesta propagació que transporta les dades de la fase és realitza en dues parts: anada i tornada.

Durant l'anada el missatge transporta diverses dades d'un mòdul: el seu valor potencial, el valor mínim i qui és el pare. A la tornada s'obté el valor màxim.

- El **valor potencial** és un nombre enter que indica la distància que hi ha fins el màster fent un recorregut en profunditat antihorari per l'arbre. Aquest inicia a zero i es va incrementant a mida que passa per un mòdul nou.

- El **valor mínim** és una de les dues dades necessàries per identificar una branca. Aquest valor és igual al valor potencial més petit dels mòduls que formen part de la branca, és a dir, indica el valor potencial del pare de la branca. Aquesta dada es manté igual a tots els mòduls de la branca.

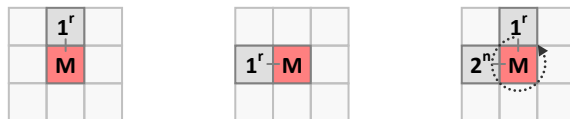
- El **valor màxim** és l'altra dada (juntament amb el valor mínim) necessària per identificar una branca. Aquest valor és igual al valor potencial més gran dels mòduls que formen part de la branca, és a dir, indica el valor potencial de la fulla de la branca. Aquesta dada es manté igual a tots els mòduls de la branca.
- El **pare** és el valor que identifica en quin dels quatre punts cardinals es troba el pare del mòdul (vegeu la Figura 5.1).



**Figura 5.1: Identificació del pare d'un mòdul.** Es mostra com s'identifica un pare en funció d'on es troba respecte a un mòdul concret (mòdul blau). Si el pare del mòdul es troba al Nord el seu pare és l'1, si és a l'Oest el seu pare és el 2, si està a l'Est és el 3, i si es troba al Sud és el 4.

El màster és qui fa el paper d'arrel de l'arbre i és per on comença el procés de construcció de la funció potencial.

El mòdul màster inicia el procés d'**anada**. Per les propietats d'aquest tipus de mòdul (definides a l'Apartat 2.2) només pot estar connectat pel Nord o per l'Oest. En cas d'estar connectat a totes dues branques, l'ordre s'estableix amb prioritat Nord (vegeu la Figura 5.2). També inicialitza el seu mínim a zero, ja que ell és el mòdul més petit de la branca actual. I com que el màster no té pare, manté el valor que fa referència a aquesta dada a zero.



**Figura 5.2: Propagació del missatge d'un mòdul màster.** Es mostra per quin mòdul propaga el missatge. En cas de tenir més d'una possibilitat la preferència és pel Nord.

La resta de mòduls es mantenen a l'espera fins que reben un missatge d'un altre mòdul amb la informació. Aquest missatge prové del seu pare, per tant, guarden aquesta dada (qui és el seu pare), i amb la informació rebuda coneixen el seu valor potencial i el valor mínim identificatiu de la branca.

Un cop extretes les dades, propaguen el missatge al mòdul següent. A diferència del màster, la resta de mòduls determinen per quina direcció ha de continuar en funció de per on venia el missatge. D'entre tots els mòduls veïns als quals estan connectats, sempre l'envien al fill següent en sentit antihorari (vegeu la Figura 5.3). Si no hi ha cap veí per on continuar, vol dir que el missatge ha arribat al final d'aquella branca (el mòdul és una fulla de l'arbre) i per tant ha de fer la tornada.



**Figura 5.3: Propagació del missatge d'un mòdul.** Es mostra l'ordre que les regles segueixen per decidir per on propagar el missatge. L'ordre ve determinat en funció de per on ha arribat el missatge (fletxa vermella). L'ordre és en sentit antihorari (fletxa negra).

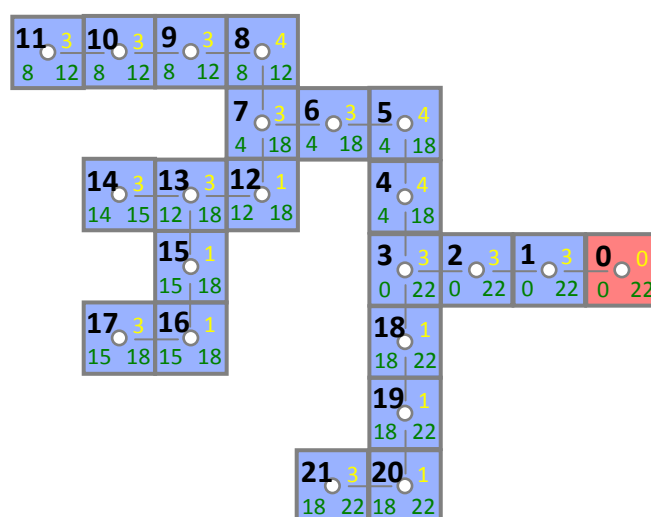
La **tornada** s'inicia quan el missatge arriba a la fulla d'una branca. Llavors, la informació que es propaga és: el valor potencial, que es mantindrà igual fins que es trobi un nou mòdul pel qual encara no ha passat; i el valor màxim, que identifica el final de la branca i, ara que es coneix, es pot transmetre durant el recorregut de tornada a la resta de mòduls que formen part d'aquella branca. Els mòduls pels quals retorna el missatge ja disposen de totes les dades i, per tant, poden fer el canvi d'estat per passar a la fase següent.

Al llarg de la construcció, els mòduls es van marcant a mida que circula per ells el missatge (d'anada) de la funció potencial. Això ho fa servir un mòdul que té diverses branques per determinar per quines ha passat prèviament el missatge. D'aquesta manera, un mòdul pot saber si encara li queda alguna branca per la qual continuar fent el procés d'anada o, en canvi, ha de fer el procés de tornada.

El mòduls en els quals es bifurquen diverses branques (tenen 3 o 4 mòduls veïns enganxats) són els encarregats de variar els valors mínims que identifiquen una nova branca. Això es fa en propagar un missatge d'anada a una nova branca enviant el nou valor mínim (igual al valor potencial del mòdul al qual li està enviant). Aquests mòduls no passen a fer el procés de tornada fins que no han completat totes les seves branques, i és llavors quan actualitzen el seu màxim i el propaguen al seu pare.

El procés de construcció de la funció potencial finalitza quan el missatge de tornada arriba al màster i no existeixen més branques per les quals propagar el missatge o, el que és el mateix, quan tots els mòduls del robot han obtingut totes les dades (valor potencial, valor mínim, valor màxim i posició del pare).

La Figura 5.4 mostra un exemple de com queden les dades un cop finalitzada la fase de construcció de la funció potencial.



**Figura 5.4: Exemple amb les dades de la fase finalitzada.** Es mostren totes les dades obtingudes un cop finalitzada la fase bpf. En negre el valor potencial, en groc la posició del pare, en verd el mínim i vermell el màxim.

## 5.3 Regles

Les regles per dur a terme aquesta fase estan subdividides en 7 grups. El primer grup inclou les regles que inicien el procés; el segon, les que envien el missatge per primer cop a un mòdul; el tercer, les que envien el missatge per primer cop però s'han d'esperar per poder continuar; el quart, les regles que continuen després de l'espera de les anteriors; el cinquè grup és aquell que continua amb el missatge per mòduls pels que ja havia passat; el sisè, les que fan la tornada enrere del missatge; i el setè, les que finalitzen el procés en arribar al màster de nou.

Les regles d'aquesta fase s'identifiquen per les sigles [bpf] (*Build Potential Function*) al llarg del camí d'anada, i per [pfc] (*Potential Function Complete*) durant la tornada. Els tres primers caràcters de l'estat d'un mòdul es fan servir per la identificació de les dues parts d'aquesta fase, el quart caràcter serveix per diferenciar en quin estat es troba dintre de la fase, i l'últim per diferenciar el mòdul màster i el líder de forat de la resta de mòduls. Per tant, identifiquem els mòduls de la forma següent:

[bpf*M]	Màster en la fase build potential function
[bpf*H]	Líder de forat en la fase build potential function
[bpf_]	Mòduls en espera del missatge de valor potencial
[bpfm]	Mòduls fent el recorregut de marcatge (anada)
[bpfw]	Mòduls fent el recorregut de marcatge (en espera)
[pfc_]	Mòduls fent el recorregut de marcatge (tornada)
[pfcA]	Mòduls fulla amb intenció d'activar-se

Per veure la descripció detallada de les regles, vegeu l'Annex B.

### 5.3.1 Grup: “start”

Aquest grup està format per 3 regles, que contemplen les tres situacions possibles en les quals es pot trobar el màster: que estigui connectat pel Nord, connectat per l'Oest, o per tots dos alhora. Les regles indiquen al màster per on ha d'iniciar la propagació del missatge en cada cas.

Per iniciar el procés de construcció de la funció potencial, els mòduls veïns del màster han d'haver completat la fase anterior de construcció de l'arbre.

Quan el màster aplica alguna de les regles d'aquest grup, inicialitza les dades (valor potencial i valor mínim) i les envia al fill corresponent. Un cop completat, el mòdul queda marcat.



### 5.3.2 Grup: “*first go*”

Aquest grup està format per 24 regles subdividides en 4 subgrups de 6 regles. Cadascun dels subgrups fa referència a cadascuna de les 4 coordenades des d'on pot arribar el missatge de la funció potencial (N, S, E, W). Cadascun d'aquests subgrups té 3 regles d'un tipus, i 3 regles d'un altre.

El primer tipus tracta els casos en què el missatge recorre la mateixa branca. L'altre, aquells casos en què la propagació del missatge es troba amb una bifurcació on, per tant, continuarà per una nova branca.

Cada tipus té tres regles diferents per determinar quina direcció ha de prendre. Per fer això, cal mirar a quins veïns està connectat el mòdul i que tots ells hagin superat la fase anterior. Cada combinació de connexions comporta un resultat diferent.

Totes aquestes regles formen part de l'anada de la construcció de la funció potencial. Al llarg d'aquesta anada es recorren parts de l'arbre que encara no han estat “marcades”.

Aquestes regles actualitzen el valor potencial, el valor mínim i determinen qui és el pare del mòdul. També envien el valor potencial al primer fill i, en funció del tipus de regla aplicada, envien el mateix valor mínim identificatiu de la branca o el nou. Aquest mòdul, finalment, queda marcat.

### 5.3.3 Grup: “*first wait*”

Aquest grup està format per 4 regles, una per cadascun dels 4 punts cardinals per on pot arribar el missatge de funció potencial (N, S, E, W).

Aquestes regles tracten els casos en què un mòdul no pot continuar propagant el missatge perquè algun dels mòduls veïns encara es troba en una fase prèvia (no pot aplicar cap de les regles “*first go*”), per tant, el mòdul es manté en estat d'espera.

Com la informació que es propaga no es pot perdre, el mòdul en espera ha de guardar totes les dades necessàries per continuar transmetent el missatge tan bon punt sigui possible.

Les regles d'aquest grup emmagatzemen les dades que li arriben al mòdul i la seva procedència en comptadors. El mòdul passa a l'estat d'espera “*wait*”.

#### **5.3.4 Grup: “wait-continue”**

A l'igual que el grup “first go”, està format per 24 regles subdividides en 4 subgrups de 6 regles.

La similitud amb el grup 2 (“first go”) es deguda a què aquestes regles realitzen la mateixa funció però, en aquest cas, a partir de mòduls que es troben en estat d'espera. Aquest fet comporta que les dades ja no arribin a través d'un veí, sinó que s'agafin dels comptadors on aquesta informació està emmagatzemada.

Aquestes regles apliquen quan tots els mòduls veïns es troben en disposició de continuar amb la propagació del missatge (han superat la fase anterior). Llavors, el mòdul envia la informació al mòdul que pertoca continuant amb el camí d'anada de la construcció. El mòdul deixa d'estar en estat d'espera i passa a estar “marcat”.

#### **5.3.5 Grup: “continue”**

Aquest grup està format per 8 regles subdividides en 4 subgrups de 2 regles. Cada subgrup fa referència a cadascuna de les 4 coordenades des d'on pot arribar el missatge de la funció potencial (N, S, E, W). En cada cas, tracta els dos possibles veïns per on continuar amb la propagació.

Aquestes regles s'apliquen a mòduls que tenen bifurcacions. Són mòduls pels quals ja ha passat el missatge, ha tornat a ells, però encara queden branques per les quals no l'ha propagat.

Cada subgrup contempla només dos camins perquè els altres dos han d'estar ja marcats necessàriament: el del pare i, com a mínim, el d'un fill previ. Com que el missatge continua per un camí pel qual encara no ha circulat, aquest grup de regles segueix realitzant el camí d'anada de la construcció de la funció potencial.

Els mòduls que apliquen aquestes regles ja tenen les dades de valor potencial, valor mínim i qui és el seu pare perquè no és el primer cop que passa el missatge per ells. Aquestes regles, doncs, envien el valor potencial que ha arribat des de l'últim mòdul marcat (la fulla de la branca anterior) al fill següent, i el nou valor mínim identificatiu de la nova branca. El mòdul que aplica la regla ja estava marcat prèviament i, com encara li queda camí per on propagar el missatge, el seu estat no varia.

### 5.3.6 Grup: “*last*”

Aquest grup està format per 12 regles subdividides en 4 subgrups de 3 regles. Cada subgrup tracta una de les 4 coordenades per on pot arribar el missatge (N, S, E, W). Cadascun d'aquests subgrups contempla els tres possibles camins on es pot trobar el pare del mòdul. El quart camí (no contemplat) correspon a la tornada del missatge pel mateix lloc per on ha arribat.

Les regles d'aquest grup realitzen el camí de tornada de la construcció de la funció potencial.

Els mòduls que apliquen aquestes regles ja disposen de totes les dades necessàries per poder passar a la fase següent. Actualitzen el seu valor màxim i propaguen aquest valor al seu pare. Un cop finalitzat, el seu estat passa al de fase completada [pfc] (*Potential function complete*).

### 5.3.7 Grup: “*last father*”

Aquest grup està format per 2 regles.

Aquestes regles només s'apliquen al màster i contemplen els dos camins pels quals li pot arribar el missatge (N, W).

Només es pot aplicar una d'aquestes regles quan el missatge ha recorregut tot l'arbre, ha retornat al màster i, per tant, la construcció de la funció potencial està completa i la fase finalitzada.

En cas que el missatge arribi al màster pel Nord, la regla ha de verificar que no tingui un segon camí per l'Oest pel qual propagar el missatge. Si el missatge li arriba per l'Oest no necessita cap altra comprovació perquè, tot i tenint un mòdul al Nord, aquest camí s'haurà recorregut prèviament degut a l'ordre de propagació que segueix el màster.

Aquestes regles fan que el màster obtingui el seu valor màxim, que correspon també al nombre total de mòduls que formen el robot. Finalment, el màster passa a l'estat de fase completada [pfc].

### **5.3.8 Grup extra: regles adaptades (“first go adapt” i “first wait adapt”)**

Aquest grup està format per les mateixes 24 + 4 regles que els seus grups anàlegs (“first go” i “first wait”).

Aquestes regles tenen les mateixes condicions i realitzen la mateixa funció amb els mateixos resultats que les seves regles equivalents. La diferència és que els mòduls que apliquen aquestes regles es troben en una fase anterior, un estat intermedi que permet visualitzar en el simulador la circulació dels missatges.

Així doncs, aquestes regles no són necessàries estrictament per a la construcció de la funció potencial, sinó tan sols per a la seva visualització gràfica al simulador.

## **5.4 Problemes**

A continuació expliquem els diferents problemes que han sorgit a mida que s’implementaven noves regles i quina solució hem adoptat.

**5.4.1 Ordre dels fills d’un mòdul:** el fill pel qual un mòdul ha de propagar el missatge no segueix sempre el mateix ordre, varia depenent de per on li ha arribat el missatge al mòdul.

*Solució adoptada: hem separat les regles en funció de per on pot arribar un missatge a un mòdul. Partint d’aquesta base hem fet, per cada cas, tota la combinatòria de connexions possibles.*

**5.4.2 Repetició del recorregut:** en fer la propagació del missatge de valor potencial és necessari saber per quins mòduls hem passat prèviament. Les raons són dues: determinar si estem fent el camí d’anada o de tornada de la construcció del valor potencial, i no repetir dos cops un mateix camí.

*Solució adoptada: decidim fer servir un dels caràcters de l’estat d’un mòdul perquè qualsevol altre el pugui identificar com a mòdul “marcat”. El quart caràcter de l’estat determina si és un mòdul normal [\*\*\*\_\*], marcat [\*\*\*m\*], o en espera [\*\*\*w\*].*

**5.4.3 Conflicte d'estats:** fer el marcatge d'un mòdul produeix que un dels seus caràcters es faci servir per aquesta tasca, però es necessari que aquest caràcter quedi lliure per altres fases. El problema és que en treure aquesta marca el mòdul veí pensa que per aquell mòdul encara no ha passat el missatge i, per tant, li torna a enviar.

*Solució adoptada: fer servir la dada del pare i el fet que la propagació d'un missatge segueix l'ordre antihorari. Amb aquesta informació podem dir que si seguim l'ordre antihorari i ens trobem amb el pare del mòdul, és segur que ja hem donat tota la volta i visitat la resta de mòduls veïns. D'aquesta manera no és necessari mantenir el marcatge del mòdul un cop s'està realitzant la tornada.*

**5.4.4 Marcatge d'un fill:** la idea és que el mòdul que envia les dades ha de marcar el fill al qual les envia per no repetir el mateix camí. El problema és que no disposem d'una funcionalitat capaç de modificar l'estat d'un altre mòdul. Un mòdul només pot fer canvis del seu propi estat.

*Solució adoptada: el pare envia el missatge i és el fill qui, al rebre'l, es marca modificant el seu propi estat.*

**5.4.5 Informació extra, identificació de branques:** necessitem identificar si dos mòduls pertanyen a una mateixa branca, però no disposem de cap informació que les diferenciï.

*Solució adoptada: volem fer servir els valors potencials dels mòduls que es troben en els límits de les branques per diferenciar unes de les altres. Aprofitem que aquesta fase realitza dos camins, anada i tornada. A l'anada identifiquem el mòdul amb el valor més petit d'una branca (el màster o el primer després d'una bifurcació) i propaguem aquest valor fins arribar al final de la branca o a una bifurcació. En arribar al final d'una branca, ens trobem amb el mòdul fulla. Aquest mòdul també és el que té el valor potencial més gran d'aquella branca, per tant, agafem el seu valor potencial i el propaguem a la tornada de la fase per tota la branca (fins arribar a una bifurcació o al màster).*

**5.4.6 Regles de "continue" que no apliquen mai:** hi ha casos (en el grup de regles de "continue") que mai no es poden donar. Dels 4 possibles veïns a considerar per un mòdul, un d'ells és aquell que li ha enviat el missatge, i un altre és el seu pare. Per tant, només pot haver un màxim de dos camins viables (els dos següents seguint l'ordre antihorari).

*Solució adoptada: eliminem aquelles regles que per les propietats del sistema sabem que mai no poden aplicar ja que aquests casos pertanyen al grup de regles "last".*

**5.4.7 La funció OR no existeix:** en diverses regles necessitem tenir en compte diferents alternatives en una mateixa precondition, però la funcionalitat per fer una OR lògica no existeix al simulador.

*Solució adoptada: el simulador no disposa de ORs però, en canvi, sí hi ha parèntesi “( )” per agrupar condicions i signe d’exclamació “!” per realitzar una NOT i negar un valor lògic. A més, (per defecte) totes les condicions estan separades per una AND. Fent servir aquestes tres funcionalitats es pot obtenir el resultat equivalent en fer una OR. L’equivalència lògica d’una OR és la negació de cada condició, i la posterior negació del conjunt. Per exemple: “x OR y” equival a “!( !x !y )”.*

**5.4.8 Creuament de fases:** un mòdul aplica les regles tan bon punt li és possible fer-ho, sense importar que encara hi hagi altres mòduls en fases anteriors. Això pot provocar que diversos mòduls treballin en fases diferents. Quan un mòdul que transporta el valor potencial es troba amb veïns que encara no han superat la fase [bst] (construcció de l’arbre), pot prendre la decisió errònia d’enviar el missatge a un mòdul que posteriorment es desconnectarà i, per tant, aquell valor potencial no correspondrà a l’arbre que finalment s’ha generat. També es pot donar el cas que un mòdul pensa que té diverses branques i envia un nou valor mínim per identificar la nova branca, quan en realitat només en té una i, per tant, el valor mínim s’hauria de mantenir igual.

*Solució adoptada: abans que un mòdul envii les dades, per evitar anar per un camí erroni, s’ha de cerciorar que tots els mòduls involucrats es troben en la fase adient (en aquest cas la fase bpf). Hem afegit aquesta comprovació dels mòduls veïns a les regles de propagació del missatge. També hem implementat noves regles que mantenen les dades del mòdul a l’espera que tots els veïns estiguin llestos per poder continuar amb la propagació del missatge. Les regles que mantenen les dades a l’espera pertanyen al grup “first wait” (vegeu l’Apartat 5.3.3), i les encarregades de reprendre la propagació són les del grup “wait-continue” (vegeu l’Apartat 5.3.4).*

**5.4.9 Missatge de marca:** les regles d’aquesta fase fan servir el missatge de text “mark” per comunicar a un mòdul que li està enviant dades de construcció de la funció potencial i que s’ha de marcar si encara no ho està, però el simulador només disposa d’un únic canal per enviar un missatge de text. Aquest canal també es fa servir per enviar el missatge d’un candidat de màster o de líder de forat, per tant, es produeix un conflicte si s’envien dos missatges alhora.

*Solució adoptada: com que el missatge “mark” és més simple que els missatges dels candidats (el text “mark” és merament simbòlic), el substituïm per un missatge numèric. D’aquesta forma deixem lliure l’únic canal de comunicació de text, i fem servir un dels canals numèrics lliures per enviar el missatge de marca.*

**5.4.10 Conflictes d'adaptació amb les regles de *Master* i *Hole*:** són diversos problemes comentats prèviament, relacionats amb l'adaptació entre aquesta fase i les fases de cerca de màster i detecció de forats (vegeu els apartats 2.4 i 3.4).

Deixant de banda problemes ocasionats per la limitació de funcionalitats al simulador i algun petit problema propi de la fase, la major part de les complicacions han vingut donades pel control de conflictes entre fases o la necessitat d'informació no prevista. La solució a aquestes complicacions ha incrementat notablement la quantitat i complexitat de les regles d'aquesta fase.

## 5.5 Modificacions

Quan s'implementa aquesta fase, el màster i els líders de forat iniciaven ja detectats perquè s'identificaven de forma manual, això implicava que tots els mòduls arribaven a aquesta fase de forma sincronitzada. Aquest fet simplificava les regles ja que no era necessari cap tipus de control de fases. Tots els mòduls podien aplicar les seves regles sense preocupar-se de l'estat dels mòduls veïns.

Donat que no existia la possibilitat que hi hagués diversos mòduls treballant en diferents fases, i que cada fase s'aplicava de forma seqüencial, no hi havia necessitat d'aplicar regles de forma paral·lela, per tant, es podia treure profit de les prioritats per simplificar les regles.

Un cop s'inclouen les regles que automatitzen aquest procés manual, el canvi de fases es produeix de forma asíncrona. Per resoldre aquest inconvenient, s'implementen noves regles que controlen aquesta col·lisió de fases. Aquestes regles són aquelles que pertanyen als grups "*first wait*" i "*wait-continue*". D'altra banda, s'unifiquen les prioritats per permetre que les dues fases puguin treballar en paral·lel. Aquesta decisió provoca que les precondicions de les regles siguin més complexes, però el sistema sigui més eficient.

D'altra banda, en un primer moment no vèiem la necessitat d'identificar les branques d'un arbre. Això canvia amb la implementació de la fase següent (vegeu l'Apartat 6.2.2). Per tal d'aconseguir aquesta informació, separem les regles per controlar si un missatge s'envia per la mateixa branca, o hi ha una bifurcació i el missatge passa a un mòdul que forma part d'una nova branca.

Finalment, per solucionar el problema del solapament de colors esmentat en els apartats 2.4.5 i 3.4.5, es dupliquen les regles dels grups "*first go*" i "*first wait*" per adaptar els casos en els quals un missatge de valor potencial arriba a un mòdul en fase [ini] i ha de transmetre el missatge i fer el canvi a l'estat que li pertoca.

## **5.6 Alternatives**

No s'ha plantejat cap alternativa al sistema per realitzar la construcció de la funció potencial, però sí al procés d'obtenció d'algunes dades concretes:

Per saber si un mòdul està marcat fem servir un dels caràcters del seu estat.

Una altra opció és fer servir una fase intermèdia en la qual tots els mòduls comuniquen el seu estat als seus veïns. Aquesta opció es descarta en entendre que seria necessari que tots els mòduls fessin tot el procés de forma sincronitzada i tot el sistema es faria més lent, ja que necessitarien el doble de temps per realitzar cada acció. Una alternativa a aquesta, però amb resultat similar, és que només els mòduls no marcats fossin aquells que comuniquessin el seu estat als veïns.

Abans d'incloure les regles que realitzen la cerca del màster i la detecció de forats, es feien servir les prioritats per donar preferència a unes regles sobre unes altres. D'aquesta manera ens estalviàvem haver de considerar tots els casos possibles a les precondicions de cada regla. Un cop s'inclouen aquestes regles, es decideix prescindir de les prioritats per treballar en paral·lel.

Una alternativa és mantenir l'ordre de prioritats i separar totalment les fases de cerca de màster i detecció de forats de les de construcció de l'*scan tree* i construcció de la funció potencial. D'aquesta forma es redueix la complexitat de les precondicions, però es necessita implementar noves regles en les que, un cop detectats el màster i els líders de forat, aquests comuniquin a la resta de mòduls que ja poden passar a construir l'arbre. Aquest procés no és trivial.

Una forma de separar completament les fases és que el màster pugui enviar a tots els mòduls un missatge en el mateix instant de temps, però no disposem d'aquesta facilitat. L'opció més similar és que el màster faci circular un missatge de propagació en amplada per tot l'arbre.

Enviar un missatge en amplada té uns resultats similars a l'alternativa adoptada (deixar que actuïn tan bon punt puguin) amb dues petites diferències. La primera és que trigaria més en actuar (menys eficient) perquè tot mòdul ha d'esperar que el missatge arribi al màster, i que li torni a arribar a ell. I l'altra és que, quan arriba el missatge al màster, es pot donar per finalitzada la fase actual i començar la següent, fent que el canvi entre fases sigui més fàcil de veure per l'usuari.



## 5.7 Models de prova

Els jocs de prova d'aquesta fase es poden dividir en dos grups: els realitzats quan màster i forats es marcaven manualment, i els realitzats un cop la detecció es fa mitjançant les regles.

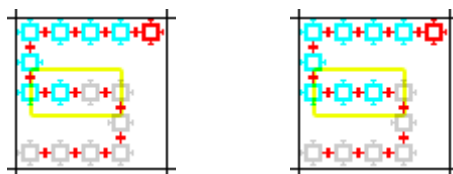
Els primers verifiquen el correcte funcionament en la propagació del missatge: increment dels valors, identificació de branques, camí escollit en una bifurcació, marcatge dels mòduls i diferenciació del camí d'anada amb el de tornada.

Els segons comproven que no hi ha conflictes ocasionats per la inclusió de les fases encarregades de trobar el màster i de detectar els forats.

En aquest apartat analitzarem només el primer grup, ja que els que fan referència a conflictes entre fases han estat analitzats en apartats anteriors (vegeu els apartats 2.6.2, 2.6.4, 3.6.2 i 3.6.3).

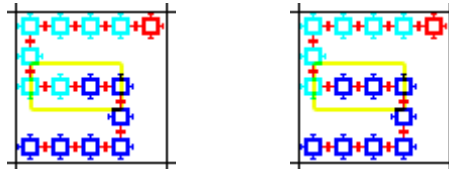
Tot i que també hem fet servir models complexos per verificar el correcte funcionament d'aquesta fase, en els exemples tractem models més simplificats per facilitar la comprensió visual de les principals comprovacions fetes.

**5.7.1 Propagació del missatge d'anada:** durant la propagació del missatge d'anada s'envia un missatge de marca, el valor potencial, i el valor mínim. A la Figura 5.5 s'observa com el mòdul blau propaga el missatge al mòdul següent, el mòdul gris. En aquest procés el mòdul blau (de valor potencial 7) envia el missatge, i el gris modifica els seus valors: el valor potencial s'incrementa (a 8), el valor mínim es manté igual que el del blau (a 0) perquè continua a la mateixa branca, s'identifica el pare (West), es "marca", i propaga el missatge al mòdul següent.



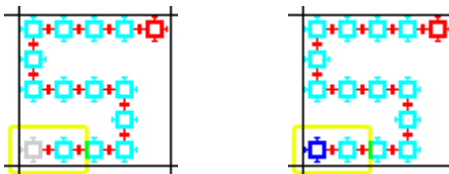
**Figura 5.5: Missatge d'anada.** A la primera imatge el mòdul blau envia el missatge d'anada al mòdul gris. A la segona imatge el mòdul gris ara és blau perquè ha estat marcat. Ara és aquest mòdul qui envia el missatge al següent mòdul gris. Els mòduls grisos representen aquells que es troben sense marcar [bpf\_], i els blaus els marcats [bpfm] que representen el camí d'anada.

**5.7.2 Propagació del missatge de tornada:** al llarg de la propagació del missatge de tornada, s'envia un missatge de marca i el valor màxim. En aquest cas, a la Figura 5.6 s'observa com el mòdul blau fosc propaga el missatge al mòdul següent, el mòdul blau cel. Ara és el mòdul blau fosc qui envia el missatge, i el blau cel qui modifica els seus valors: el valor màxim es manté igual que el del blau fosc (a 15) perquè continua a la mateixa branca, canvia d'estat per indicar que ha completat la fase i disposa de totes les dades, i propaga el missatge al mòdul següent (en aquest cas, al seu oest).



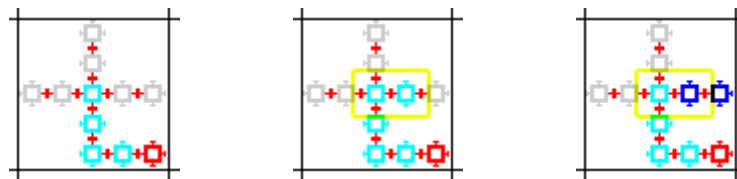
**Figura 5.6: Missatge de tornada.** A la primera imatge el mòdul blau fosc envia el missatge de tornada al mòdul blau cel. A la segona, com que el mòdul que era blau cel ja té totes les dades de la fase, canvia a blau fosc. Ara és ell qui continua amb la propagació del missatge de tornada. Els mòduls blau cel representen aquells que han estat marcats [bpfm], els blau fosc els que han completat la fase [pfc\_] i representen el camí de tornada.

**5.7.3 Propagació del missatge a una fulla:** quan un missatge arriba a la fulla d'una branca es comporta de forma diferent. Ha de fer les tasques que realitzen el camí d'anada i de tornada alhora. A la Figura 5.7 es pot observar que passa directament a l'estat de fase completada. El mòdul fulla, en rebre el missatge, modifica totes les seves dades: incrementa el seu valor potencial (a 14), identifica el seu pare (*East*), el valor mínim el manté igual que el del seu pare (a 0), determina el valor màxim de la branca (15), canvia el seu estat, i propaga el missatge de tornada cap al seu pare.



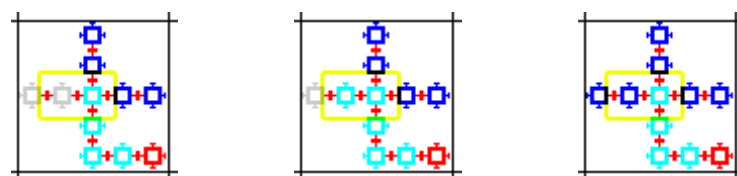
**Figura 5.7: Missatge a una fulla.** A la primera imatge el mòdul blau cel envia el missatge d'anada al mòdul gris (la fulla). A la segona, com que el mòdul que era gris disposa de totes les dades de la fase, canvia directament a blau fosc. Ara el mòdul fulla comença a propagar el missatge de tornada. El mòdul gris representa aquell que encara no està marcat [bpf\_], els blau cel aquells ja marcats [bpfm], i el blau fosc el que ha completat la fase [pfc\_].

**5.7.4 Bifurcació per la primera branca:** s'han realitzat experiments per tractar totes les combinacions de direcció per les quals pot arribar un missatge, i varietat de branques per les quals pot bifurcar-se. A la Figura 5.8 tractem el cas en què una branca que prové del Sud es separa en tres subbranques. El mòdul que es troba a la bifurcació ha de determinar per quina branca enviar el missatge en primer lloc. Com es pot observar, segueix correctament l'ordre antihorari establert. Quan una branca es bifurca, els valors mínims i màxims de la branca han de variar. En el cas de la Figura 5.8, els valors de la branca principal són: mínim 0 i màxim 11; en canvi, els de la primera subbranca són: mínim 5 i màxim 7. D'aquesta manera és pot determinar que tot i ser branques diferents, una penja de l'altra.



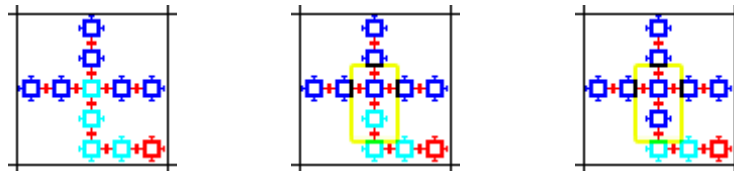
**Figura 5.8: Triple bifurcació, primera branca.** A la primera imatge el mòdul central es troba amb tres branques per les quals propagar el missatge. A la segona, el missatge segueix per la branca de la dreta (la primera en ordre antihorari). A la tercera, quan el missatge retorna, es torna a trobar la bifurcació. Els mòduls grisos representen aquells que encara no estan marcats [bpf\_], els blau cel aquells ja marcats [bpfm], i els blau fosc els que han completat la fase [pfc\_].

**5.7.5 Bifurcació per l'última branca:** fem servir el mateix model que en el cas anterior per veure com continua la propagació del missatge un cop completades les branques anteriors (Figura 5.9). El mòdul que es troba a la bifurcació ha de determinar per quina branca anar. En aquest cas, només li queda una branca sense marcar, per tant, propaga el missatge per la branca que li resta. De nou, els valors mínims i màxims de la branca han de variar. En aquest cas, els valors de l'última subbranca són: mínim 9 i màxim 11. Podem extreure de les dades obtingudes que l'última subbranca també penja de la branca principal però, en canvi, no té cap relació amb la resta de subbranques.



**Figura 5.9: Triple bifurcació, darrera branca.** A la primera imatge el mòdul blau cel es troba amb tres branques, la de la dreta completada, la de l'esquerra sense marcar, i la de sota marcada (la de dalt és d'on li arriba el missatge). A la segona, el missatge segueix per la branca de l'esquerra, la que queda sense marcar (i la següent en ordre antihorari). A la tercera, quan el missatge retorna, es torna a trobar la bifurcació, aquest cop amb totes les branques marcades. Els mòduls grisos representen aquells que encara no estan marcats [bpf\_], els blau cel aquells ja marcats [bpfm], i els blau fosc els que han completat la fase [pfc\_].

**5.7.6 Tornada d'una bifurcació:** fem servir el mateix model que en els dos casos anteriors per veure com continua la propagació del missatge un cop completades totes les branques (Figura 5.10). El mòdul que es troba a la bifurcació ha de determinar per quina branca anar. En aquest cas totes les branques ja estan marcades, per tant, el camí d'anada s'ha completat i només queda fer la tornada per la branca principal (la que correspon a on es troba el pare). El mòdul on es bifurquen totes les branques ja pot modificar el seu valor màxim (11) i propagar-lo per la branca principal.



**Figura 5.10: Triple bifurcació, tornada.** A la primera imatge el mòdul central es troba amb tres branques, la de la sota marcada però no completada, i les altres marcades i completades (la de l'esquerra és d'on li arriba el missatge). A la segona, com que el mòdul de la bifurcació ha propagat el missatge a totes les subbranques, passa a estar també complet i canvia el seu color a blau fosc. A la tercera s'observa com el missatge retorna per la branca de sota, l'única que queda sense completar. Els mòduls blau cel representen aquells que han estat marcats [bpfm], i els blau fosc els que han completat la fase [pfc\_].

## 6 Reconfiguració a la forma canònica

### 6.1 Objectiu

L'objectiu dels mòduls que es troben en aquesta fase és la de desfer la forma inicial del robot de manera que passi a tenir una forma canònica. Aquest procés es pot dividir en dues fites generals: arribar al màster i generar la tira.

La primera s'aconsegueix fent que els mòduls circulin per la frontera exterior del robot mitjançant moviments simples (desplaçar-se d'un mòdul a un altre, girar al voltant d'un mòdul o connectar-se a un altre mòdul veí) fins trobar-se amb el màster. Aquests moviments han d'evitar les col·lisions i obstruccions amb altres mòduls.

La segona fita és, un cop el mòdul ha trobat el màster, generar la forma canònica. Aquesta forma és l'agrupació de tots els mòduls enganxats un darrere l'altre formant una tira. La forma canònica es genera a l'Est del mòdul màster. Cada mòdul que arriba al màster s'ha de desplaçar sobre la tira fins arribar al final i així poder situar-se.

La fase de reconfiguració a la forma canònica finalitza un cop tots els mòduls formen part de la tirallonga.

### 6.2 Estratègia

A l'inici d'aquesta fase l'arbre està construït, el màster assignat i els mòduls tenen informació sobre el seu valor potencial, la branca en la qual es troben i la posició del seu pare.

La reconfiguració a la forma canònica la dividim en tres parts: activació, moviment i adhesió.

#### 6.2.1 Activació

L'activació només la realitzen els mòduls que són fulles de l'arbre. La forma d'identificar una fulla és pel seu nombre de connexions. Exceptuant el màster, tot mòdul que només està connectat per una de les quatre coordenades és una fulla.

A mida que els mòduls actius es van movent es desconnecten dels seus pares. Una vegada que un mòdul pare s'ha desfet de tots els fills, passa a ser fulla. Finalment tots els mòduls (a excepció del màster) seran fulles en algun moment de la reconfiguració i, per tant, tots els mòduls s'acabaran activant.

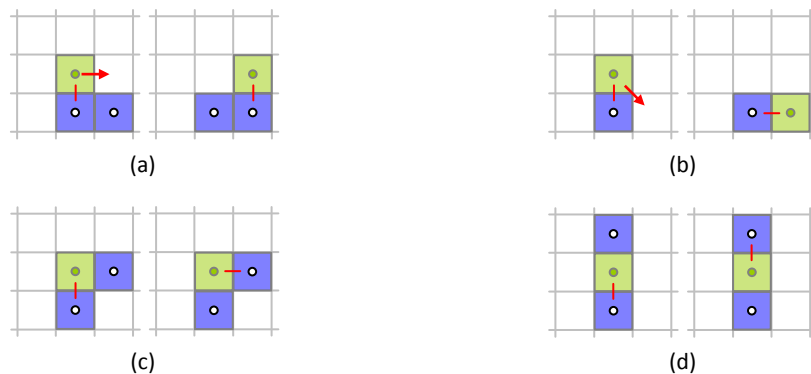
L'activació es realitza en dos temps, la primera per anunciar la seva intenció d'activar-se, i la segona per realitzar el canvi d'estat a actiu. Això es fa per fer saber a possibles mòduls que volen moure's al damunt d'un altre que aquest té la intenció d'activar-se i s'han d'esperar.

Tot mòdul actiu es manté actiu fins que no finalitza la fase (s'enganxa a la tira). Aquest tipus de mòduls sempre estan connectats a un únic veí estàtic (no actiu).

### 6.2.2 Moviment

El moviment el realitzen només mòduls actius i sempre sobre mòduls estàtics. Els tipus de moviments que pot realitzar un mòdul són quatre: *Slide*, *Convex Transition*, *Concave Transition* i *Opposite Transition*.

- **Slide** consisteix en desconnectar-se del mòdul al qual està connectat, moure's a un altre adjacent i connectar-se a aquest nou mòdul (vegeu la Figura 6.1.a).
- **Convex Transition** es tracta d'un moviment en diagonal, equival a girar una cantonada convexa per posicionar-se a un altre costat del mateix mòdul. El mòdul es desconnecta, es mou a una altra cara del mòdul al qual estava enganxat i es connecta de nou al mateix mòdul. Per realitzar aquest moviment només és necessari que la posició de destí estigui lliure, el mòdul actiu no necessita espai extra per moure's (vegeu la Figura 6.1.b).
- **Concave Transition** no produeix cap moviment de posició però sí un canvi de connexió. El mòdul es desconnecta del seu veí per connectar-se al mòdul adjacent a la següent de les seves cares. El canvi de situació equival a girar una cantonada còncava (vegeu la Figura 6.1.c).
- **Opposite Transition** com en el cas "concave", no produeix cap moviment i sí un canvi de connexió. En aquest cas, el mòdul es desconnecta del seu veí per connectar-se al veí que es troba al costat oposat (vegeu la Figura 6.1.d).



**Figura 6.1: Moviments d'un mòdul.** Es mostra l'estat previ i posterior a cada moviment. (a) Slide. (b) Convex Transition. (c) Concave Transition. (d) Opposite Transition. El verd representa un mòdul actiu i el blau a un mòdul estàtic.

Tots els moviments d'un mòdul segueixen 3 pautes principals: utilitzen la regla de la mà dreta, es mouen sobre mòduls estàtics i el moviment el fan sobre mòduls amb valor potencial més petit.

**La regla de la mà dreta** es fa servir per assegurar que un mòdul no passa dos cops pel mateix camí (mateixa posició i connexió) sense haver-se trobat prèviament amb el màster. Seguint aquesta regla, un mòdul que es troba a l'interior del robot acabarà sortint i recorrent la part externa, que és on es troba l'arrel de l'arbre (el màster). Entenem aquesta mà dreta com la connexió del mòdul i no com la dreta de la configuració general. Per exemple, en la Figura 6.1.d abans de realitzar el moviment la seva dreta està enganxada al mòdul del Sud i la seva intenció és anar en direcció Est o Sud-est, en canvi, un cop ha realitzat l'*Opposite Transition* la seva dreta està enganxada al mòdul Nord i la seva intenció és anar en direcció Oest o Nord-oest.

El moviment sempre es realitza **sobre mòduls estàtics** per evitar que el moviment d'un dels mòduls deixi a un altre desconnectat del robot.

Per altra banda, fer que el moviment sempre **disminueixi el valor potencial** ens assegura que amb cada moviment estiguem més a prop del màster. Hi ha una excepció a aquesta pauta explicada més endavant.

Tenint en compte l'estructura de l'arbre del robot i l'assignació feta dels valors potencials dels mòduls, en aplicar aquestes pautes aconseguim que l'arbre es vagi desfent seguint un ordre jeràrquic. Primer es desfan les subbranques més properes al màster (menor valor potencial) i segueix cap a les més llunyanes fins a desfer-les totes. Un cop desfetes totes les subbranques, es desfan les branques d'on penjaven, i així, fins a desfer tots els mòduls que pegen del màster. Entenem per desfer una branca com la desconnexió i moviment en direcció al màster de tots els mòduls que formen part d'aquella branca.

Els mòduls actuen de forma independent, de manera que podem trobar diversos mòduls movent-se de forma asíncrona i simultània. Aquest fet pot produir que mòduls actius d'una part de l'arbre s'acumulin a la seva arrel esperant que la branca anterior acabi de desfer-se per deixar el camí lliure i poder reprendre el camí. De la mateixa manera, aquesta asincronia pot ocasionar diversos problemes que hem de controlar.

Descrivim breument els possibles problemes (es troben detallats a l'Apartat 6.4):

- **Col·lisions:** té lloc quan dos mòduls es volen moure a una mateixa posició (vegeu l'Apartat 6.4.2).
- **Obstruccions:** té lloc quan dos mòduls es troben un al costat de l'altre i cadascun vol anar en direcció oposada (vegeu l'Apartat 6.4.3).
- **Culs de sac:** entrar i sortir d'un cul de sac pot produir les col·lisions i obstruccions anteriors. Quan un mòdul entra en un cul de sac pot produir un conflicte amb un mòdul que surt (vegeu els apartats 6.4.4 i 6.4.5). Per veure diferents formes de culs de sac, vegeu la Figura 6.2.

Part dels moviments d'un mòdul també es fan en **dos temps**: un per anunciar quin tipus de moviment vol fer i l'altre per dur-lo a terme. Amb això s'eviten les col·lisions entre mòduls. Els moviments afectats són aquells que impliquen desplaçament (*Slide* i *Convex Transition*), que són els que poden produir una col·lisió. Hi ha un cas especial d'*Opposite Transition* que també fa servir dos temps però, en aquest cas, per obtenir dades extra de les que no disposa (per més detalls vegeu l'Apartat 6.4.7).

Tot mòdul que vol realitzar un moviment ha de verificar prèviament que tots els mòduls que poden afectar a la seva decisió es troben en la fase actual (tenen totes les dades que necessita per decidir quin és el moviment òptim).

En cas que es pugui produir un conflicte, el mòdul que té preferència és aquell amb el valor potencial més petit. Per tant, de tots els mòduls implicats, només aquell que té un menor valor potencial realitza l'acció; la resta, tornen al seu estat anterior a l'espera d'una nova oportunitat de realitzar el seu moviment.

El moviment natural dels mòduls és desplaçar-se per la seva branca fins arribar a la seva arrel i continuar per les branques de les que penja fins arribar al màster. Per optimitzar aquest procés, els mòduls tenen la capacitat de saltar a branques menors a la seva. Això serveix tant per avançar a branques més properes al màster, com per evitar entrar dintre d'un cul de sac.

La forma de decidir si és millor saltar o continuar endavant no es trivial. Hi ha diverses formes de branques i depenent de en quin cas es troba el mòdul, ha d'actuar una manera o d'una altra.

La idea general és que un mòdul sempre ha d'evitar entrar en un cul de sac, si es dona aquest cas ha de saltar. El problema és com identificar si el cas en el qual es troba el mòdul és un cul de sac i si està entrant o sortint.

Els casos possibles de **cul de sac** que es poden donar són 128, com a combinació de:

- 2 tipus de sac: de cantonada o de front.
- 2 tipus de branca: mateixa branca o branques diferents.
- 2 tipus de valor potencial: més gran o més petit.
- 4 formes de coll d'ampolla: interna-interna, interna-externa, externa-interna, externa-externa.
- 4 coordenades possibles a les quals està connectat el mòdul: Nord, Sud, Est, Oest.

Aquí només analitzarem els casos de sacs de front i d'una de les coordenades, ja que els altres casos són anàlegs.



- **Mòdul enganxat a una branca diferent amb valor potencial més gran** (Figura 6.2.a):

Com el moviment del mòdul segueix la regla de la mà dreta, en els quatre casos va en direcció Est i està entrant al cul de sac. El valor potencial de la branca oposada és més petit que el de la branca en la qual es troba. El mòdul ha de saltar (*Opposite Transition*) en tots els casos.

- **Mòdul enganxat a una branca diferent amb valor potencial més petit** (Figura 6.2.b):

En els quatre casos el mòdul va en direcció Oest i està sortint del cul de sac. El valor potencial de la branca oposada és més gran que el de la branca en la qual es troba. El mòdul ha de continuar endavant en tots els casos (*Slide* en els casos de l'exemple).

- **Mòdul enganxat a la mateixa branca amb valor potencial més gran** (Figura 6.2.c):

En aquest cas, tot i que saltant el mòdul milloraria el seu valor potencial, no sempre és la millor opció. En el primer i segon cas (c1 i c2) el mòdul va en direcció Oest i està entrant al cul de sac, per tant, la millor opció és saltar (*Opposite Transition*). En canvi, en el tercer i quart cas (c3 i c4) el mòdul va en direcció Est i està sortint del cul de sac així que és millor continuar endavant (*Convex Transition*).

Per diferenciar els casos c1 i c2 dels c3 i c4 no és suficient amb l'anàlisi del valor potencial del mòdul al qual podríem saltar, també és necessari analitzar les dades dels seus adjacents. Sabent com estan situats els seus adjacents i on es troben els pares d'aquests, podem determinar a quin costat de la branca es troba el màster i, per tant, saber quina és la forma que té la branca.

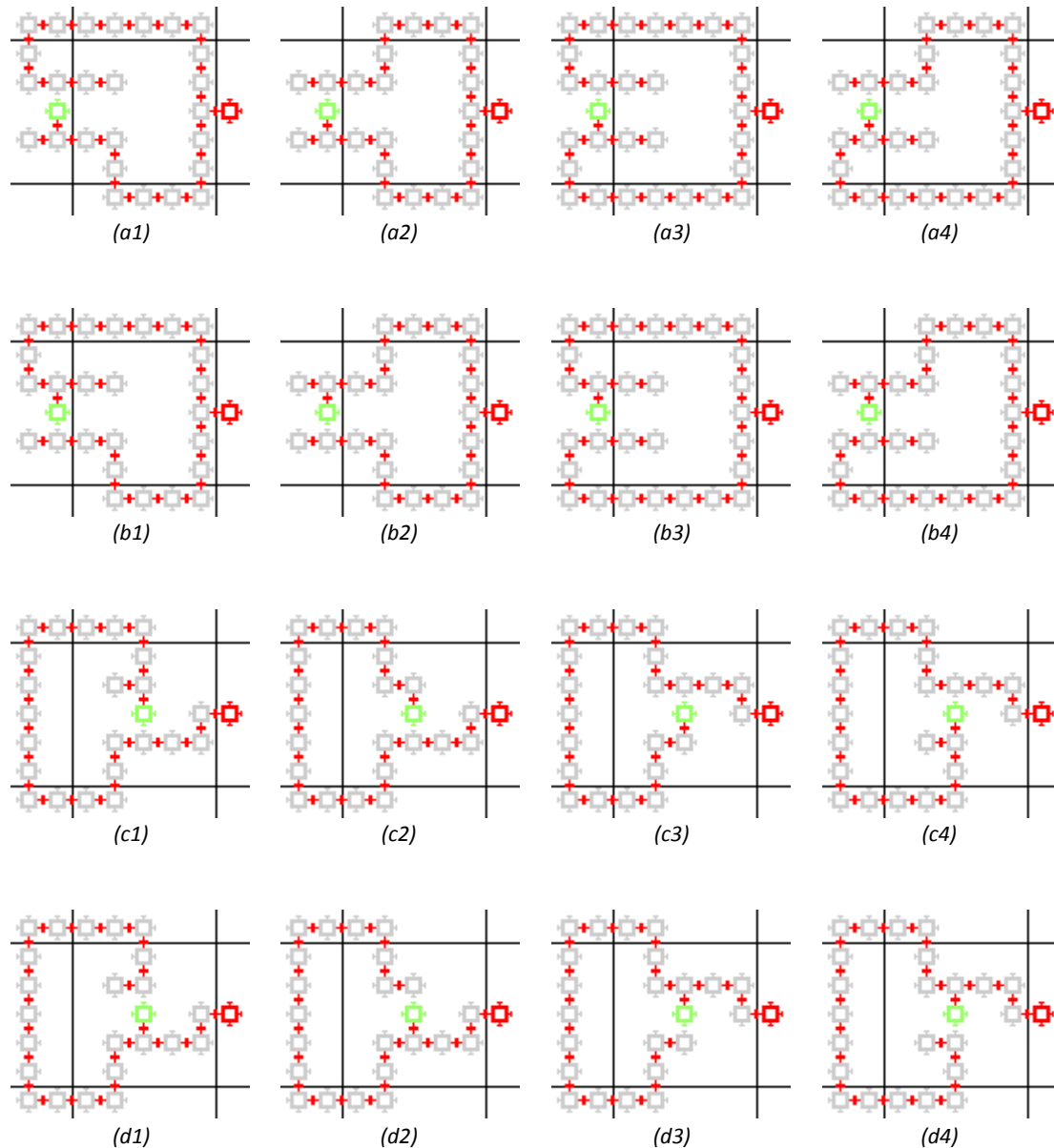
Pel valor potencial sabem que saltar situa el mòdul més a prop del màster, però no és lo mateix trobar-se a l'exterior de la branca que a l'interior. Si es troba a l'interior, el mòdul encara haurà de recórrer tota la branca fins donar-li la volta per arribar al màster. Per aquesta raó el mòdul només ha de saltar si en fer-ho es situa a la part externa de la branca (com en els casos c1 i c2).

- **Mòdul enganxat a la mateixa branca amb valor potencial més petit** (Figura 6.2.d):

En aquest cas saltar empitjoraria el valor potencial del mòdul però, tot i així, depenent de la situació pot ser millor saltar. En el primer i segon cas (d1 i d2) el mòdul va en direcció Est i està sortint del cul de sac, per tant, la millor opció és continuar endavant (*Slide*). En canvi, en el tercer i quart cas (d3 i d4) el mòdul va en direcció Oest i està entrant al cul de sac així que és millor saltar (*Opposite Transition*).

Tal i com passava en el cas anterior, per diferenciar els casos no és suficient amb el valor potencial, però aquest cop les dades extra que es necessiten són les dels adjacents del mòdul al qual està connectat. En aquests casos les dades són accessibles però no podem operar amb elles en el mateix torn, per això es necessari primer agafar les dades i després decidir si saltar o continuar. Aquesta excepció s'explica a l'Apartat 6.3.5.

Un cop obtingudes les dades necessàries la decisió és la mateixa que en l'anterior però, en aquest cas, la part de la branca analitzada és on ens trobem connectats. Si ens trobem a l'exterior de la branca continuem endavant sinó, saltem.



**Figura 6.2: Formes possibles de mòduls entrant/sortint de culs de sac.** Es mostra el mòdul verd (actiu) en les possibles situacions d'entrar o sortir d'un cul de sac. (a) Branques diferents, valor potencial més gran. (b) Branques diferents, valor potencial més petit. (c) Mateixa branca, valor potencial més gran. (d) Mateixa branca, valor potencial més petit.

### 6.2.3 Adhesió

Un cop un mòdul es troba amb el màster, sap que la forma canònica es genera al seu Est. El primer mòdul en arribar s'enganxa a l'Est del màster i es torna estàtic de nou, és el primer mòdul de la tira (sense comptar el màster). Els mòduls següents en arribar han de recórrer la tira generada i anar-se enganxant al final.

L'últim mòdul actiu que passa pel màster és aquell que tanca la tirallonga i dóna per finalitzada la fase de reconfiguració a la forma canònica. El màster pot identificar quin és aquest últim mòdul. Quan el màster té un mòdul actiu enganxat al seu Nord però no en té cap enganxat al seu Oest, és segur que ja no queda cap altre mòdul per passar.

### 6.2.4 Exemple global de l'estratègia

Per finalitzar, descrivim el que passa en algunes de les iteracions de l'exemple de la Figura 6.3. En aquest exemple es mostren els moviments que realitzen els mòduls d'un robot per arribar fins el màster i completar la forma canònica. Només es mostren iteracions concretes en les quals succeeix alguna cosa remarcable.

Iteració 106: inici de la fase de reconfiguració a la forma canònica.

Iteració 107: primera fulla que vol activar-se (mòdul groc).

Iteració 108: la fulla es converteix en un mòdul actiu (verd clar).

Iteració 109: el mòdul actiu (verd fosc) té la intenció de moure's al Nord-Est.

Iteració 110: el mòdul actiu (verd clar) realitza un *Convex Transition* al Nord-Est seguint la regla de la mà dreta.

Espera que el mòdul del Nord estigui a la fase actual per poder decidir el moviment següent.

Iteració 125: el mòdul actiu (verd fosc) ja pot anunciar que vol moure's a l'Est.

Iteració 126: el mòdul actiu (verd clar) realitza un *Slide* a l'Est.

Iteració 127: un nou mòdul (groc) vol activar-se ara que ja no té més connexió que la del seu pare.

Iteració 131: el mòdul actiu (verd clar) torna a esperar-se fins que el mòdul de l'Est estigui a la fase actual.

Iteració 144: un nou mòdul (groc) vol activar-se.

Iteració 147: el mòdul actiu (verd clar) es troba entrant en un cul de sac.

Iteració 148: el mòdul actiu (verd clar) realitza un *Opposite Transition* al Nord evitant el cul de sac i saltant a una branca anterior a la seva.

Iteració 164: el mòdul actiu (verd fosc) de sota té la intenció de moure's al Nord-Est i altre mòdul actiu (verd fosc) de sobre té la intenció de moure's a l'Oest. Tots dos volen ocupar el mateix espai.

Iteració 165: dels dos mòduls actius (verd clar) en conflicte, el de sobre té un valor potencial més petit. Per tant, realitza l'*Slide* a l'Oest, mentre que el de sota es manté a l'espera.

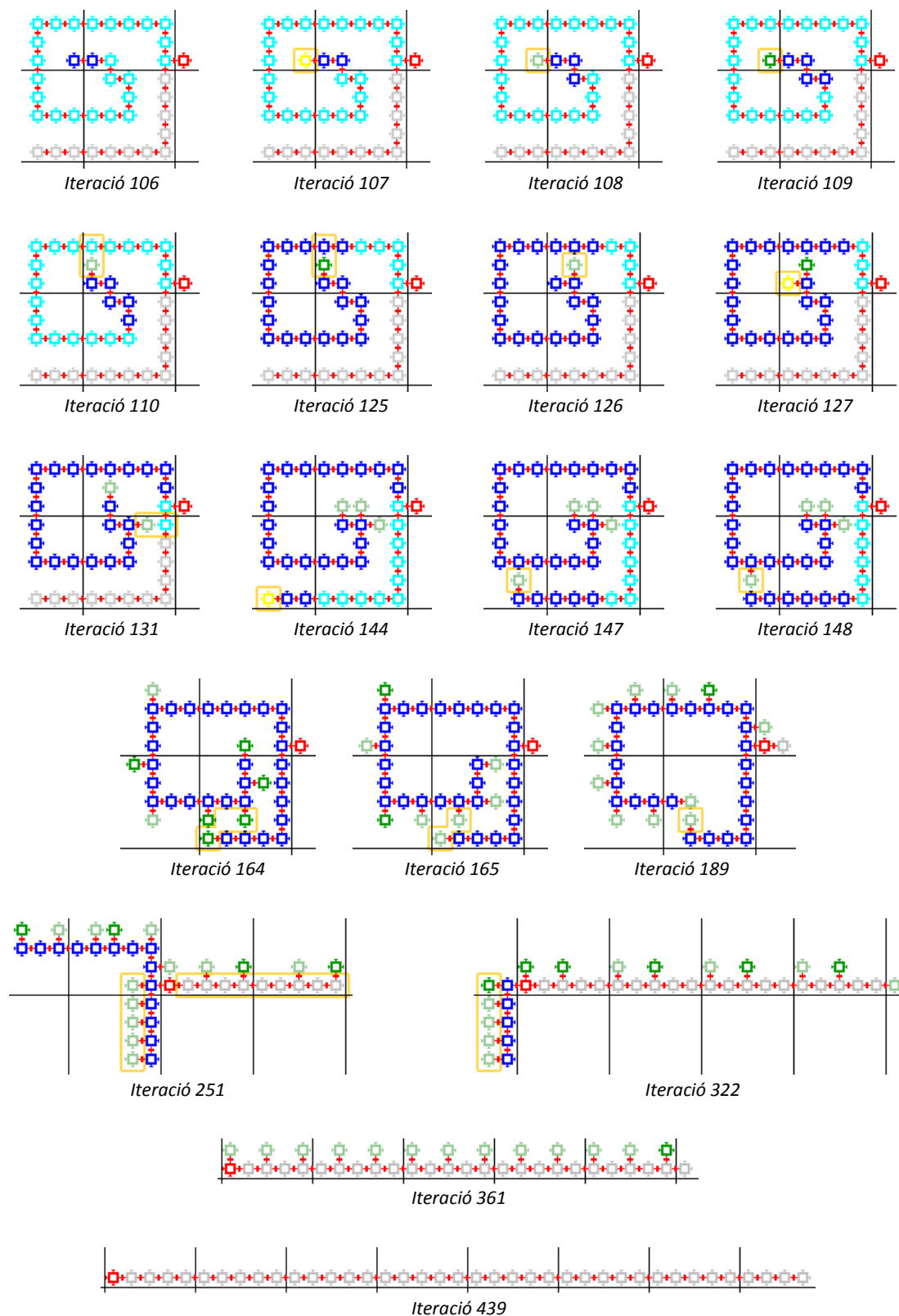
Iteració 189: el mòdul actiu (verd clar) realitza el moviment al Nord-Est ara que cap altre mòdul no li ho impedeix.

Iteració 251: els mòduls actius de la branca inferior (verd clar) es mantenen a l'espera que es desfaci l'altra branca per poder continuar. Mentrestant, els mòduls que han anat arribant al màster han anat formant la tirallonga (mòduls grisos).

Iteració 322: un cop la branca superior s'ha desfet completament, ara els mòduls actius que estaven a l'espera tenen el camí lliure per continuar.

Iteració 361: tots els mòduls han passat pel màster, ja només queda que la resta de mòduls actius (verd) s'adhereixin a la tira.

Iteració 439: finalitzada la fase, el robot ha completat la reconfiguració a la forma canònica. Tots els mòduls formen part de la tira.



**Figura 6.3: Exemple de reconfiguració a la forma canònica.** Es mostren diferents iteracions del procés de reconfiguració.

## 6.3 Regles

Les regles per dur a terme aquesta fase estan subdividides en 6 grups, un d'activació, quatre de moviment, i un més d'adhesió. El primer grup inclou les regles que passen els mòduls d'estàtics a actius; el segon, el grup que realitza el moviment d'*Slide*; el tercer, el moviment de *Convex Transition*; el quart, les regles que fan el moviment de *Concave Transition*; el cinquè grup és el que aplica el moviment d'*Opposite Transition*; i el sisè, és aquell que s'encarrega de les adhesions dels mòduls a la tirallonga i de la identificació de l'últim mòdul actiu.

Les regles d'aquesta fase s'identifiquen per les sigles [fw] (*Forwards*). Realitzen el procés des que els mòduls s'activen fins que han finalitzat la reconfiguració. Aquest cop fem servir els dos primers caràcters de l'estat d'un mòdul per a la identificació de la fase, ja que utilitzem els dos següents per identificar la intencionalitat d'un mòdul. L'últim caràcter està reservat per diferenciar el mòdul màster, els mòduls actius i els mòduls que han finalitzat la fase.

Hi ha dues excepcions, una per identificar un mòdul amb la intenció d'activar-se, i una altra pels mòduls que han finalitzat el procés. La primera fa servir els tres caràcters de la fase [pfc] perquè el mòdul encara pertany a la fase anterior i un quart caràcter 'A' (*Activate*) per indicar la seva intenció. La segona fa servir els mateixos tres caràcters de la fase [pfc] amb un últim caràcter 'E' (*End*), la raó d'aquesta última excepció s'explica a l'Apartat 6.4.8. Per tant, identifiquem els mòduls de la forma següent:

[pfcA]	Mòduls fulla amb intenció d'activar-se
[fw__A]	Mòduls actius en espera
[fwN_A]	Mòduls actius amb intenció de moure's al Nord
[fwS_A]	Mòduls actius amb intenció de moure's al Sud
[fwE_A]	Mòduls actius amb intenció de moure's a l'Est
[fwW_A]	Mòduls actius amb intenció de moure's a l'Oest
[fwNEA]	Mòduls actius amb intenció de moure's al Nord-Est
[fwSEA]	Mòduls actius amb intenció de moure's al Sud-Est
[fwNWA]	Mòduls actius amb intenció de moure's al Nord-Oest
[fwSWA]	Mòduls actius amb intenció de moure's al Sud-Oest
[fwdcA]	Mòduls actius decidint el camí a seguir (saltar o continuar)
[pfc_E]	Posicionat a la tira (End)

Per veure la descripció detallada de les regles, vegeu l'Annex C.

A diferència de les fases prèvies, un mòdul no pot aplicar les regles d'aquesta fase simultàniament a regles de fases anteriors. Com hem comentat anteriorment, les prioritats determinen amb quina preferència s'aplica una regla sobre la resta. Totes les regles d'aquesta fase tenen una prioritat menor a la de les fases anteriors. Dintre d'aquesta fase, cada grup de regles tenen un ordre de prioritat general que determina quin tipus de moviment aplica primer i un ordre particular per diferenciar les parts en els quals es divideix cada grup.

**L'ordre general de prioritat** és el següent: els grups amb prioritat més gran són els d'*Activate i Strip* (inici i final de la fase), el primer perquè un mòdul no es pot moure fins que no s'ha activat, i el segon perquè no continuï movent-se un cop ha arribat al final; seguidament es troben l'*Opposite Transition* i *Concave Transition*, que són els moviments capaços d'avançar major distància en poder saltar els trams que provoquen els culs de sac; i per últim estan els moviments simples d'*Slide* i *Convex Transition*, que són els que fan que els mòduls es desplacin pas a pas.

**L'ordre particular de prioritats** dins cada grup pot ser: per diferenciar els dos temps (primera i segona iteració), per diferenciar les parts (intencionalitat, verificació i acció), per diferents tipus de branca (diferent, mateixa, millot, pitjor), o per tractar casos especials (quan la resta de regles no apliquen). Algunes de les diferenciacions de prioritats no són estrictament necessàries.

Nota: durant el procés s'emmagatzemen dades que no són realment necessàries per aconseguir l'objectiu però que serveixen com a informació extra dels mòduls. Una dada és el "Signe", que inicialment es va fer servir com a verificació del moviment. L'altra dada són les "Coordenades" relatives d'on es troba un mòdul respecte al seu punt d'origen.

### **6.3.1 Grup: "activate"**

Aquest grup està format per 9 regles, dividides en 2 subgrups de 4 regles més una regla especial. El primer subgrup tracta la intencionalitat del mòdul i el segon realitza l'activació. Cadascuna de les regles del subgrup fa referència a cadascuna de les 4 coordenades on es pot trobar la fulla de la branca (N, S, E, W). La regla especial anul·la la intenció del mòdul tornant a l'estat anterior.

Per iniciar el procés de reconfiguració a la forma canònica, abans de començar a moure's, un mòdul s'ha d'activar. Un mòdul s'activa si és una fulla i, tant ell com el seu pare, ja tenen tota la informació (han completat la fase anterior [pfc]). Una fulla és aquell mòdul que, sense ser màster, només està connectat a un altre mòdul, el seu pare.

Abans que un mòdul s'activi, ha de comunicar-ho a altres possibles mòduls en moviment. Per realitzar aquesta comunicació fem servir els estats dels mòduls. Hi ha un sub-grup de regles que s'encarrega d'anunciar la intenció d'activar-se d'un mòdul. A la iteració següent, si continua podent activar-se, l'altre sub-grup s'encarrega de canviar l'estat del mòdul a actiu. Si per alguna raó ja no pot activar-se (algun mòdul actiu s'ha mogut al seu damunt i, per tant, ha deixat de ser una fulla), la regla especial s'encarrega de tornar el mòdul a l'estat anterior.

Al llarg de tot el procés d'aquesta fase, les dades d'un mòdul actiu estan directament relacionades amb les del mòdul al qual es troba connectat. En activar-se, les seves dades s'actualitzen en funció de les que té el mòdul al qual està connectat. El seu valor potencial s'incrementa en 1 respecte el del seu pare. La branca a la qual pertany, ara és la mateixa que la del seu pare. Per tant, els valors que la identifiquen passen a ser també els mateixos. Aquestes dades, tot i que normalment són les que el mòdul ja tenia, varien si el seu pare és el mòdul d'una bifurcació.

Quan un mòdul s'activa, inicialitza les seves coordenades a (0,0).

### 6.3.2 Grup: “slide”

Aquest grup està format per 28 regles subdividides en 4 subgrups de 7 regles. Cadascun dels subgrups fa referència a cadascuna de les 4 coordenades cap on el mòdul vol realitzar el moviment (N, S, E, W). Cadascun d'aquests subgrups té 1 regla d'intencionalitat, 5 regles de verificació i 1 regla d'acció.

*Slide* és un dels quatre tipus de moviments que pot fer un mòdul, aquest consisteix en desplaçar-se des d'una posició a una contigua. Dit d'una altra forma, només modifica una de les coordenades del mòdul, tot incrementant-la o decrementant-la en 1.

Tan sols els mòduls actius poden realitzar un moviment d'*Slide*, però per poder fer-ho hem de verificar l'estat del seu entorn. Per tal que un mòdul pugui fer el moviment d'*Slide* s'han de complir diverses condicions: tots els mòduls que poden afectar la decisió del moviment s'han de trobar en l'estat de [pfc] (tenir tota la informació assignada); la posició on el mòdul vol anar ha d'estar lliure i cap mòdul millor l'ha de voler ocupar; i moure's a aquella posició ha de millorar el seu valor potencial. Considerem que un mòdul és millor quan el seu valor potencial és més petit i, per tant, les seves accions tenen preferència.

Per aconseguir que es compleixin aquestes condicions es segueixen tres passos: intencionalitat, verificació i acció.

La **intencionalitat** la mirem en una primera iteració. El mòdul fa una comprovació bàsica per determinar que pot realitzar el moviment i canvia el seu estat per comunicar el tipus de moviment que vol fer a la resta de mòduls.

En la segona iteració realitzem la **verificació**, en la qual hi ha les regles per comprovar cadascuna de les 5 causes que poden evitar el moviment del mòdul: si alguna de les comprovacions bàsiques ja no es compleix, si el mòdul al qual volem anar es vol activar o si algun dels 3 mòduls possibles (amb un valor potencial més petit) vol ocupar la posició on volem anar.

Si cap de les regles de verificació s'aplica, vol dir que el mòdul no té cap impediment per realitzar l'**acció** de moviment. L'acció es realitza en la mateixa iteració que la verificació, per tant, el procés triga dos "torns".

En completar el moviment, el mòdul actualitza les seves dades amb les del nou mòdul al qual s'ha enganxat. Aquestes dades són el seu valor potencial i els valors que representen la seva branca. També actualitza, en funció del tipus de moviment, les seves coordenades relatives.

### **6.3.3 Grup: "*convex transition*"**

Aquest grup està format per 32 regles subdividides en 4 subgrups de 8 regles. Cadascun dels subgrups fa referència a cadascuna de les 4 coordenades cap on el mòdul vol realitzar el moviment (NE, SE, NW, SW). Cadascun d'aquests subgrups té 1 regla d'intencionalitat, 6 regles de verificació i 1 regla d'acció.

*Convex Transition* és un altre dels quatre tipus de moviments que pot fer un mòdul. Aquest moviment consisteix en desplaçar-se des d'una posició a una contigua en diagonal, o el que és el mateix, modifica les dues coordenades incrementant-les o decrementant-les en 1. En aquest cas el moviment es realitza al voltant del mateix mòdul al qual es troba enganxat, connectant-se al mateix mòdul però en un altre dels seus costats.

Tal com passa amb l'*Slide*, només els mòduls actius poden fer el moviment de *Convex Transition*, i per poder fer-ho han de revisar prèviament el seu entorn.

Per tal que un mòdul pugui fer el moviment de *Convex Transition*, s'han de complir les mateixes condicions que en el cas anterior (*Slide*). L'única diferència és que en aquest cas hi ha més mòduls involucrats en la decisió de moviment i que el seu valor potencial no millora sinó que es manté igual. Tot i que conceptualment el moviment que realitza és com girar al voltant d'un altre mòdul, només necessita l'espai físic de la posició de destí per realitzar aquest tipus de moviment.



Igual que amb l'*Slide*, per aconseguir que un mòdul compleixi aquestes condicions, es segueixen els mateixos tres passos: intencionalitat, verificació i acció.

La **intencionalitat**, que fa la comprovació bàsica i canvia el seu estat; la **verificació**, per comprovar, en aquest cas, les 6 causes que poden evitar el moviment; i, en el cas que cap regla no ho eviti, la realització de l'**acció** del moviment.

En aquest cas, en realitzar el moviment, el mòdul només ha d'actualitzar les seves coordenades. Com segueix connectat al mateix mòdul, les dades del seu valor potencial i la seva branca es mantenen iguals.

#### 6.3.4 Grup: "*concave transition*"

Aquest grup està format per 12 regles subdividides en 3 subgrups de 4 regles. El primer subgrup tracta els casos on els dos mòduls veïns implicats pertanyen a branques diferents. El segon i tercer subgrups tracten els casos on els veïns implicats formen part de la mateixa branca, un subgrup per als que millora el valor potencial, i un altre per als que empitjora. Cadascuna de les 4 regles d'un subgrup fa referència a una de les 4 coordenades a les quals el mòdul pot estar connectat (N, S, E, W).

*Concave Transition* és un altre dels quatre tipus de moviments possibles. En aquest cas, el mòdul no es mou físicament però, conceptualment, el moviment equival a realitzar un gir de 45 graus. Aquest moviment es fa modificant la connexió del mòdul a un dels costats contigus.

Com la resta de moviments, el *Concave Transition* només el poden fer els mòduls actius.

Per poder realitzar el moviment de *Concave Transition*, s'han de complir un altre tipus de condicions. En no haver moviment no es necessari realitzar una comunicació d'intencionalitat prèvia. En aquest cas, la verificació que realitza el mòdul per saber si es pot moure, té en compte l'estructura de la branca en la qual es troba.

Es poden donar diversos tipus d'estructura, l'anàlisi dels quals no fem extensament aquí, a causa de la seva analogia amb els tipus analitzats en l'Apartat 6.2.2). En aquells casos els mòduls afectats estan de front i en els casos de concave transition els mòduls afectats es troben en cantonada. Amb la informació del valor potencial, les branques i els parels dels mòduls implicats, és possible determinar el tipus d'estructura i prendre la decisió òptima en cada cas.

La forma de determinar que dues branques són diferents és comparar el valor màxim de l'altra branca amb el valor potencial del mòdul. Si el valor potencial és superior al valor màxim de la branca, el mòdul no pot pertànyer a aquella branca, per tant, podem determinar que la branca en la qual es troba és diferent de la branca a la qual té intenció de connectar-se. Fent la comprovació d'aquesta forma no només sabem que les branques són diferents, sinó també que l'altra branca és anterior a l'actual (està més a prop del màster).

Determinar que dues branques són la mateixa no és tan simple com mirar que els valors de mínim i màxim siguin iguals. Dues branques poden ser la mateixa però tenir valors diferents si entre elles hi ha alguna bifurcació. Considerem que una branca que penja d'una altra és la mateixa (una pertany a l'altra). D'aquesta manera, podem dir que dues branques són iguals si el mínim i màxim d'una es troba entre el mínim i màxim de l'altra.

En el cas que siguin la mateixa branca, hem d'esbrinar la seva forma. Aquesta forma es pot deduir amb els valors potencials dels mòduls i la direcció dels seus pares. Amb aquestes dades determinem a quina part de la branca es troba el màster i a quin costat de la branca es troba el mòdul. Fet això, podem concloure si en realitzar el *Concave Transition* s'està sortint o entrant en un cul de sac i decidir si realitzar o no el moviment.

Si es realitza el moviment, el mòdul ha d'actualitzar les seves dades amb les del nou mòdul al qual s'ha enganxat. Aquestes dades són el valor potencial i els valors de la branca. En aquest cas, a causa de l'absència de moviment, les coordenades es mantenen sense canvis.

### **6.3.5 Grup: “*opposite transition*”**

Aquest grup està format per 25 regles subdividides en 4 subgrups diferents, 3 d'ells de 4 regles i 1 amb les 13 restants. El primer subgrup tracta els casos on els dos mòduls veïns oposats pertanyen a branques diferents. El segon i tercer subgrups tracten els casos on aquests mòduls formen part de la mateixa branca, un subgrup per als que millora el valor potencial i un altre per als que empitjora. El quart subgrup tracta un cas especial en el qual un mòdul necessita aconseguir dades extra. Els tres primers subgrups tenen 4 regles on cadascuna fa referència a una de les 4 coordenades a les quals pot estar connectat el mòdul (N, S, E, W). El quart subgrup està format per 3 regles per cada coordenada (N, S, E, W) més una regla global.

*Opposite Transition* és l'últim dels quatre tipus de moviments possibles. En aquest cas, el mòdul tampoc no es mou físicament però, conceptualment, el moviment equival a realitzar un gir de 90 graus. Aquest moviment el fa modificant la connexió d'un costat al seu oposat.

Igual que amb el *Concave Transition*, només poden fer el moviment d'*Opposite Transition* els mòduls actius. I, de la mateixa manera, per decidir si realitza el moviment el mòdul ha de verificar quina és l'estructura de la branca on es troba.

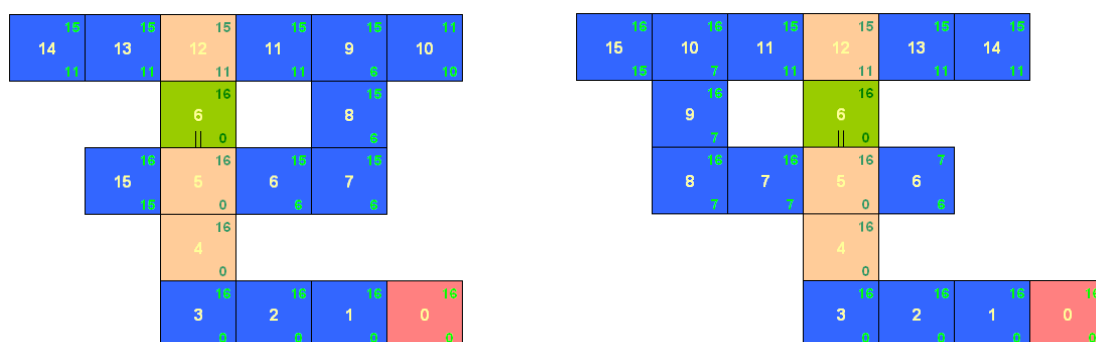
Els tipus d'estructura són anàlogues al cas anterior (comentats prèviament a l'Apartat 6.2.2). Amb la informació del valor potencial, les branques i els pares dels mòduls implicats, és possible determinar de quin tipus es tracta.

En els **casos normals**, per determinar si dues branques són diferents o la mateixa, i quina és la forma que té, utilitzem el mateix mètode que amb el *Concave Transition*.

En el **cas especial** el criteri és el mateix però el mòdul no disposa de les dades necessàries per determinar el tipus de branca. Necessitem realitzar una obtenció de dades prèvia. Per aconseguir aquesta informació, el mòdul necessita un torn per obtenir-la i un altre per decidir si pot o no aplicar l'*Opposite Transition*.

Aquest cas només es dona quan la branca és la mateixa i el valor potencial de la branca a la qual vol saltar és més gran que la del mòdul. Per determinar la forma de la branca cal comparar dades de dos mòduls diferents però, amb les funcionalitats que tenim, això no és possible. Per aquesta raó, en la primera iteració guarda en comptadors propis les dades dels dos mòduls i a la segona iteració els compara i decideix. La decisió no només depèn de la comparació, també de que el mòdul oposat no s'hagi activat ni vulgui fer-ho.

La necessitat d'agafar, en aquest cas, dades de dos mòduls diferents està motivada pel fet que el mòdul sobre el qual es troba el mòdul actiu pot ser una bifurcació i, com que totes les subbranques pertanyen a la bifurcació, amb les dades del propi mòdul (són les mateixes que les de la bifurcació) no pot determinar quina de les dues subbranques és la que es troba al costat oposat. En l'exemple de la Figura 6.4 podem observar com, tot i ser dos casos oposats, les dades que utilitzem normalment per diferenciar-los són exactament les mateixes.



**Figura 6.4: Cas especial.** Es mostren dos casos oposats de mateixa branca i valor potencial pitjor. Els valors centrals dels mòduls representen el valor potencial i els de la cantonada els mínim i màxim de la branca. El vermell representa el mòdul màster, el blau els mòduls estàtics, el taronja els mòduls estàtics dels quals s'agafa (normalment) la informació per determinar la forma, i el verd el mòdul actiu (d'on s'agafa l'altra part de la informació). En el cas de l'esquerra el mòdul actiu està entrant al cul de sac. En el cas de la dreta el mòdul actiu està sortint del cul de sac. En canvi, les dades dels mòduls "taronja" i "verd" de cada cas són idèntiques.

Si realitza el moviment, el mòdul ha d'actualitzar les seves dades amb les del nou mòdul al qual s'ha enganxat. Aquestes dades són el valor potencial i els valors de la branca. De nou, com no hi ha moviment, les coordenades es mantenen sense canvis.

### **6.3.6 Grup: “strip”**

Aquest grup està format per 3 regles de 2 tipus. Una regla per finalitzar la fase adherint-se a la tira, i dues per marcar l'últim mòdul actiu.

Per finalitzar el procés de reconfiguració a la forma canònica és necessari que hi hagi un mòdul que digui en quin moment s'acaba. Aquest mòdul el determina el màster i li comunica quan passa pel seu damunt.

La primera regla d'aquest grup s'encarrega de generar la tira, però no fa que els mòduls es moguin sobre ella (d'això s'encarreguen les regles de moviment). Aquesta regla només ha de verificar l'estat del mòdul que es troba en el seu Oest, que és des d'on es genera la tira, per saber si ha arribat al final de la mateixa i així poder adherir-se. Per saber si es troba a la tira, el mòdul del seu Oest ha de ser o bé el màster, o bé un mòdul que ja en forma part de la tira [pfc\_E].

Les altres dues regles són les encarregades de determinar quin és l'últim mòdul que finalitzarà la tirallonga. Una d'elles és la que indica al màster quan passa pel seu damunt l'últim mòdul actiu, i l'altra és la que marca a aquest mòdul. La forma d'identificar l'últim mòdul es descriu a l'Apartat 6.4.9.

Quan un mòdul s'adhereix a la tirallonga passa a ser de nou un mòdul estàtic, aquell mòdul ha finalitzat el procés. Un cop el mòdul finalitza la fase, neteja els comptadors on tenia la informació del seu valor potencial, del seu pare, i de la branca a la que pertany. Manté la informació de les coordenades relatives per possibles consultes i la marca que identifica l'últim mòdul.

## **6.4 Problemes**

A continuació expliquem els diferents problemes que han sorgit a mida que s'avançava en la implementació de noves regles i quina solució hem adoptat.

**6.4.1 Informació errònia d'un mòdul:** quan un mòdul s'activa, si el seu pare és una bifurcació, les seves dades sobre la branca no corresponen a on està enganxat. A causa d'això pot prendre decisions errònies basant-se en dades que no són les reals.

*Solució adoptada: actualitzem el valor potencial d'un mòdul en activar-se, d'aquesta manera tot mòdul actiu té des de l'inici la informació referent a on es troba connectat.*

**6.4.2 Comunicació d'intencionalitats:** per tal d'evitar **col·lisions** o moviments sobre mòduls **actius**, un mòdul necessita saber que volen fer la resta de mòduls del seu entorn. En realitzar els moviments dels mòduls de forma independent, dos o més mòduls poden decidir un moviment que els porti a la mateixa posició. Dos mòduls no poden ocupar el mateix espai perquè es produeix una col·lisió, un problema que hem de controlar.

*Solució adoptada: per evitar que més d'un mòdul es mogui a la mateixa posició, abans d'efectuar el moviment, un mòdul ha de conèixer la intenció de la resta de mòduls implicats. Un mòdul només pot conèixer d'un altre mòdul el seu estat i el valor dels seus comptadors. En aquest cas decidim fer servir dos dels caràcters d'un estat per indicar la intencionalitat d'un moviment. Per realitzar aquesta comunicació fem servir **dos torns**, un per comunicar la intenció i un altre per realitzar l'acció.*

*Amb aquest sistema es poden donar 2 casos: un mòdul comunica la seva intenció i l'altre realitza el moviment, o un mòdul comunica la seva intenció i l'altre també.*

*En el primer cas un mòdul comunica la seva intenció de moure's però, com que l'altre ha ocupat l'espai, al torn següent tornarà a l'estat previ i esperarà la propera ocasió. En el segon cas tots dos comuniquen la seva intenció alhora i, al torn següent, el que té valor potencial més petit realitza el moviment i l'altre s'espera. En canvi, mai no pot passar que dos mòduls es vulguin moure alhora. Això és perquè prèviament hauran estat en el cas en el qual tots dos comuniquen la seva intenció i el resultat d'això només pot donar que un mòdul es mou i la resta s'espera.*

*En el cas de l'activació dels mòduls, fent el procés en dos torns ens assegurem que no s'activi un mòdul quan en el torn anterior se li ha connectat un d'actiu, i que no se li connecti si el mòdul ha avisat que es vol activar.*

**6.4.3 Obstruccions:** la configuració d'un robot pot tenir qualsevol forma, dintre d'aquesta varietat de possibilitats es troben les que anomenem culs de sac. Un cul de sac és com un forat obert on, per aquesta obertura, només hi cap un únic mòdul. Tant dintre com fora del cul de sac poden haver mòduls actius en moviment. Aquest fet pot generar una obstrucció si els mòduls de dintre intenten sortir i alhora els de fora intenten entrar, provocant que es taponi l'entrada.

*Solució adoptada: hem d'evitar que els mòduls que intenten entrar al cul de sac ho facin. Per aconseguir això necessitem identificar un coll d'ampolla. Un cop aconseguit, si un mòdul detecta que està entrant en un cul de sac ha de saltar. Saltar vol dir que el mòdul passa a enganxar-se a l'altre costat, canviant la seva direcció, passant a sortir del cul de sac i estalviant-se tot el camí que hi ha per dintre. D'aquesta manera cap mòdul entra en un cul de sac, i per tant no es poden produir les obstruccions.*

**6.4.4 Identificar un coll d'ampolla:** aquesta forma, que es pot donar en qualsevol configuració d'un robot, pot provocar obstruccions i bloquejar la reconfiguració. Es necessari identificar aquest tipus d'estructura.

*Solució adoptada: un coll d'ampolla pot ser de dos tipus, de cantonada o de front. Per saber que ens trobem amb un coll d'ampolla només cal identificar qualsevol dels dos tipus. Per fer això necessitem conèixer l'ocupació de la zona on es troba el mòdul. Un de tipus cantonada tindrà (si ens imaginem una ics) una de les diagonals ocupada per mòduls estàtics i l'altra diagonal lliure o ocupada per mòduls actius. La de tipus front és un espai lliure o ocupat per un actiu entre dos espais ocupats per mòduls estàtics (vegeu un exemple de cada tipus a la Figura 6.5).*



**Figura 6.5: Tipus de coll d'ampolla.** Es mostren els dos tipus possibles de coll d'ampolla, de cantonada i de front. Els mòduls blaus són mòduls estàtics i els verds mòduls actius.

**6.4.5 Formes de les branques:** el cul de sac, a part de tenir els coll d'ampolla de dos tipus, poden tenir diferents formes. Depenent d'aquestes formes un mòdul estarà entrant o sortint del cul de sac, per aquesta raó és important poder identificar amb quina forma estem tractant.

*Solució adoptada: per conèixer la forma d'un cul de sac és necessària més informació. Necessitem conèixer si els dos costats de l'obertura pertanyen o no a una mateixa branca i, en cas de ser la mateixa, quina forma té. Un cop determinada la forma podem saber si un mòdul es dirigeix cap endins o cap enfora del cul de sac. Per realitzar aquesta diferenciació hem de comparar les dades del mòdul actiu amb les dels seus veïns estàtics i amb les dels pares d'aquests veïns.*

**6.4.6 Informació extra, identificació de branques:** necessitem identificar si dos mòduls pertanyen a una mateixa branca, però inicialment no disposàvem de cap informació que les diferenciï.

*Solució adoptada: tot i que la necessitat d'aquesta informació prové d'aquesta fase, la seva obtenció es produeix en la fase anterior. La solució adoptada, mitjançant els valors min i max, s'explica en el capítol anterior en l'Apartat 5.4.5.*

**6.4.7 Informació extra, comparació de comptadors de 2 mòduls:** hi ha un cas especial en la forma d'una branca en el qual necessitem comparar les dades de dos mòduls diferents per determinar si un mòdul està entrant o no en un cul de sac. Aquest tipus de comparació no és possible, les funcionalitats que tenim només ens permeten comparar les dades d'un altre mòdul amb les pròpies.

*Solució adoptada: decidim agafar les dades dels dos mòduls que necessitem per després realitzar la comparació. Aquesta tasca no podem realitzar-la en la mateixa iteració, per tant, hem de separar-la en dos torns. Aquesta captura de dades provoca que el mòdul es quedi durant una iteració en un estat intermedi [fwdcA] on "dc" (Data Capture) indica que està decidint el camí. Un cop pren la decisió, pot netejar els comptadors.*

**6.4.8 Caminar sobre la tirallonga:** els mòduls que es troben amb el màster han de continuar movent-se fins al final de la tirallonga per poder passar a formar-ne part.

*Solució adoptada: les regles de moviment estan fetes de manera que es mouen sobre mòduls que pertanyen a la fase [pfc\_]. Per tal d'aprofitar aquestes regles decidim que els mòduls que passin a formar part de la tira tindran l'estat [pfc\_E]. Tot i pertànyer a una fase diferent, s'aprofita els 3 mateixos caràcters d'una altra fase perquè les regles de moviment es continuïn aplicant sobre els mòduls actius que recorren la tira. La "E" (End) indica la finalització del procés de reconfiguració.*

**6.4.9 Identificació de l'últim mòdul:** per tal de finalitzar el procés de reconfiguració i una possible continuació necessitem determinar qui tanca aquesta fase, quin és el mòdul que determina que tirallonga s'ha completat.

*Solució adoptada: prenem la decisió que sigui l'últim mòdul actiu qui realitzi aquesta tasca, l'últim a col·locar-se a la tira. Per identificar l'últim mòdul ens basem en les característiques del màster. Sabem que tot mòdul penja del màster (és l'arrel de l'arbre) i que les branques que li pegen només poden ser pel seu Nord i/o pel seu Oest. També sabem que un mòdul actiu només pot estar connectat a un únic mòdul estàtic. Per tant, amb aquestes dues dades, podem assegurar que si un mòdul actiu passa pel Nord del màster i no té cap altre mòdul connectat al seu Oest, aquest mòdul actiu és l'últim que queda per passar.*

**6.4.10 Simulador:** la gran majoria de les precondicions i postcondicions de les regles d'aquesta fase necessiten funcionalitats que permetin operar amb dades d'altres mòduls. Cap de les funcionalitats del simulador satisfà aquestes necessitats.

*Solució adoptada: decidim realitzar modificacions al simulador per obtenir funcionalitats que permetin treballar amb dades de mòduls a distància. Les modificacions realitzades al simulador es troben detallades a l'Apartat 8.3.*

Durant el transcurs d'aquesta fase s'han produït gran varietat d'incidències. Les més problemàtiques han estat les relacionades amb la falta de funcionalitats que permeten fer coses que faciliten la feina. Sempre hem procurat que les funcionalitats del simulador siguin el més limitades i realistes possibles perquè el sistema sigui més fàcil d'adaptar a models reals, per aquesta raó, només algunes de les funcionalitats s'han afegit al simulador. Per altres funcionalitats s'han buscat altres alternatives. Les modificacions al simulador han suposat una feina extra no prevista, i les alternatives aplicades han incrementat la complexitat d'algunes regles.

Per altra banda, la fase per si mateixa també presenta certa dificultat. El més destacable ha estat aconseguir identificar el tipus i forma d'una branca per evitar les obstruccions, i la separació d'algunes regles en dues iteracions per aportar intencionalitat als mòduls i evitar els conflictes d'activació i de col·lisió.

## **6.5 Modificacions**

En començar a implementar ens adonem que per poder moure un mòdul necessitem comparar el seu valor potencial amb el dels mòduls als quals es pot connectar. Això és necessari per saber si s'està acostant al màster o no. El simulador no disposa de cap funcionalitat que permeti aquesta tasca així que el modifiquem per implementar un nou tipus de preconditionió.

Un cop disposem d'aquesta funcionalitat, passem a realitzar la comprovació dels valors potencials del destins possibles. Això requereix que qualsevol mòdul sobre el qual un mòdul actiu es vulgui moure no només sigui estàtic, sinó que ha de tenir el valor potencial ja assignat. Per complir aquesta restricció modifiquem les regles afegint preconditiones que verifiquin que tots el mòduls afectats tenen les dades necessàries (han completat la fase anterior).

A la primera versió de les regles implementades del grup *Slide* els mòduls mantenen els seus valors potencials durant tot el procés però, per evitar que un mòdul torni enrere, decidim que tot mòdul actiu actualitzi el seu valor potencial a mida que avança. Si no fem aquesta modificació a les regles, un mòdul podria tornar enrere per culpa que el seu valor potencial és més gran que el del camí que ja ha recorregut. Aquesta funcionalitat tampoc està disponible. Modifiquem el simulador de nou per obtenir una postcondició que ens permeti emmagatzemar en un comptador les dades del comptador d'un altre mòdul. Un cop feta la modificació al simulador, canviem les regles del grup *Slide* de manera que tot mòdul, en fer un moviment, actualitzi el valor potencial.

Un cop implementats més tipus de moviments, en fer proves per verificar el funcionament del conjunt, ens adonem de casos on un mòdul actiu no realitza el moviment que resultaria millor. La causa és que en activar-se un mòdul, si es troba en una bifurcació, les seves dades no corresponen a les del mòdul al qual està connectat.



Modifiquem les regles del grup *Activate* per fer que, un mòdul que s'activa, actualitzi les seves dades (tal com fan els mòduls en moure's).

Realitzant noves proves apareixen les col·lisions entre mòduls. Aquest problema es dona perquè cada mòdul es mou sense tenir en compte el que fa la resta. Decidim separar en dos temps totes les regles que comporten moviment de posició (*Slide* i *Convex Transition*), un per comunicar la seva intenció i un altre per realitzar el moviment o esperar. També modifiquem el control d'activació per realitzar-lo en dos passos i evitar que dos mòduls actius puguin connectar-se.

De la mateixa manera, apareixen les obstruccions i, amb elles, la necessitat de diferenciar les branques de l'arbre. Modifiquem regles de la fase anterior per aprofitar l'anada i la tornada de la construcció de la funció potencial per assignar els valors mínim i màxim d'una branca i, d'aquesta manera, aprofitar aquestes dades per identificar les branques durant els moviments dels mòduls.

Implementem les regles de la resta de moviments (*Opposite Transition* i *Concave Transition*) i sorgeix un nou problema, el "cas especial". Per resoldre aquest cas concret no ens serveixen les regles que tracten els casos generals, necessitem comparar comptadors de dos mòduls diferents. Aquest cop afegim regles específiques per aquest cas. Deixem les regles que tracten els casos generals, i afegim unes noves que realitzen la mateixa tasca però en dos temps. Les regles en dos temps utilitzen la primera iteració per agafar totes les dades necessàries i el segon per realitzar la comparació i prendre la decisió.

Fins aquest moment els mòduls iniciaven la fase de forma simultània. Per tal d'adaptar els moviments a un inici asíncron, ampliem les precondicions de totes les regles per verificar que tots els mòduls que poden influir en una decisió es troben en la fase [pfc] (amb totes les dades).

## 6.6 Alternatives

Durant aquesta fase hem pres certes decisions per a les quals hi havia altres alternatives en consideració. A continuació esmentem les més rellevants.

**L'actualització del valor potencial** l'hem realitzat mitjançant una modificació en el simulador perquè un mòdul no tenia cap forma simple de conèixer el valor potencial del mòdul al qual està connectat. Una alternativa és fer que tots els mòduls realitzin un broadcast continu als veïns amb el seu valor potencial. La complexitat està en gestionar aquest missatge en el temps. A més es pot donar el cas que un mòdul es mou sobre un altre que no fa broadcast perquè està aplicant una altra regla i es queda sense actualitzar el valor potencial.

La comprovació de la **intencionalitat d'un altre mòdul** es pot fer sense recórrer a la utilització dels dos temps. El propi mòdul pot “posar-se en la pell” del mòdul amb el qual pot tenir el conflicte i determinar ell mateix quin moviment farà. La complexitat d'aquest mètode és excessiva ja que necessitem tota la informació de cadascun dels mòduls que poden entrar en conflicte, i realitzar cadascuna de les comprovacions que faria cadascun d'aquest mòduls. Tot això en cadascuna de les regles que determinen un moviment. A més, part de la informació necessària no està disponible des d'un altre mòdul (per exemple, les connexions actuals d'un mòdul), per tant, també serien necessàries noves funcionalitats en el simulador.

Una opció que es va plantejar és la de fer una barreja de les dues alternatives. Aprofitar casos concrets en els quals no es necessiten dades inaccessibles o casos en els quals és segur que no hi ha conflicte, per fer regles específiques que poden estalviar a un mòdul fer els dos passos. Aquesta opció es va descartar pel fet que suposa massa càrrega de treball per tractar casos concrets que ja tractem amb les regles generals.

Per resoldre el **cas especial** en el qual es necessita comparar dades de dos mòduls diferents, es pot implementar una nova funcionalitat al simulador que faciliti aquesta tasca. Una precondition similar a les implementades prèviament (**V** i **W**) però que, en comptes que només el primer dels paràmetres pugui ser el comptador d'un mòdul extern, permeti que el segon també ho sigui. Per exemple, una funcionalitat adaptada de '**V**': [ V dx,dy, C \_ \_ \_ da,db, C \_ \_ \_ ] on dx,dy són les coordenades d'un mòdul i da, db les d'un altre.

A l'hora de tractar una **obstrucció** també es va plantejar una alternativa: en lloc de mirar de saltar en trobar-se amb un cul de sac, continuar endavant fins que es produeixi una obstrucció i llavors decidir quin dels dos mòduls causants ha de saltar per continuar tots dos pel mateix camí. Això pot solucionar el problema de les obstruccions però fa que el mòdul realitzi un recorregut innecessari.

## **6.7 Models de prova**

Per verificar el correcte funcionament d'aquesta fase, hem anat realitzant diferents proves a mida que implementàvem nous grups de regles. En aquest apartat només tractarem casos concrets. Un exemple del procés general de tota la fase el podeu veure a la Figura 6.3 de l'Apartat 6.2.

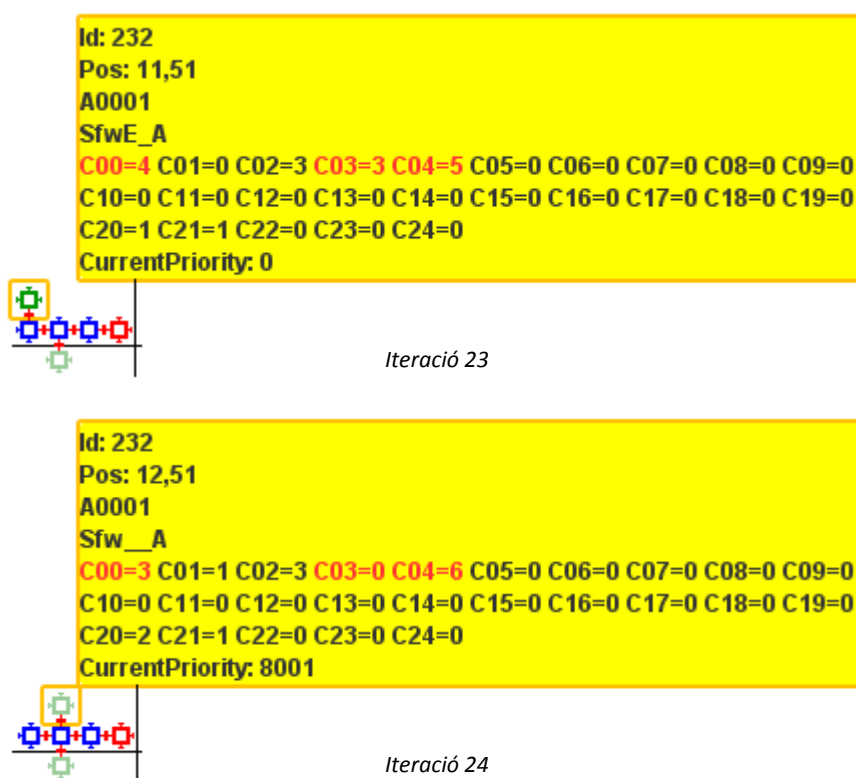
A banda dels casos que descrivim en aquest apartat, s'han realitzat proves per verificar altres casos més comuns: l'activació dels mòduls fulla; el funcionament de les regles de moviment *Slide*, *Convex Transition*, *Concave Transition* i *Opposite Transition*; l'adhesió del mòdul a la fulla; les esperes fins que els mòduls de l'entorn tinguin tota la informació; els canvis d'estat; la neteja de comptadors; i altres comprovacions més simples. La majoria d'aquests casos es poden observar en l'exemple general (Figura 6.3).

En els apartats següents mostrem només les proves realitzades per els casos que tenen complexitat més gran.

**6.7.1 Noves funcionalitats del simulador:** durant tot el procés la gran majoria de regles fan servir les noves funcionalitats. A la Figura 6.6 s'observa el correcte funcionament de les noves regles. Els mòduls actualitzen correctament els valors dels seus comptadors en funció de les dades d'uns altres mòduls.

A la iteració 23 el mòdul actiu té un valor potencial (4) i el mòdul sobre el qual es vol moure, té un valor potencial (2). Gràcies a la nova funcionalitat que permet comparar aquestes dues dades, el mòdul actiu pot realitzar el moviment d'*Slide*.

A la iteració 24 observem com el mòdul modifica el seu valor potencial i els valors de la branca amb les dades que té el mòdul estàtic. Les dades del mòdul estàtic són: valor potencial (2), mínim (0) i màxim (6).



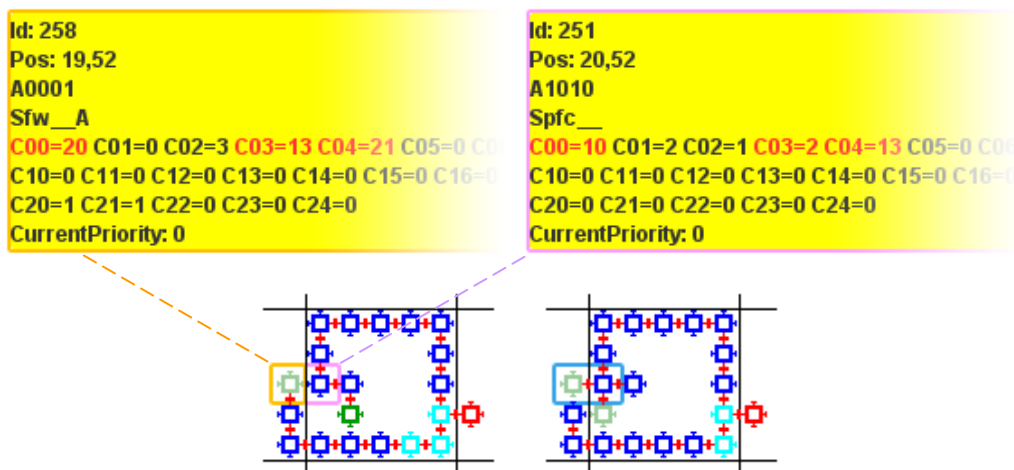
**Figura 6.6: Modificació de les dades d'un mòdul amb les dades d'un altre.** Es mostren dues iteracions consecutives d'un moviment d'*Slide*, i les dades resultants del mòdul actiu (marcat en taronja). El comptador C00 indica el valor potencial, el C03 el valor mínim i el C04 el valor màxim.

**6.7.2 Colls d'ampolla:** quan un mòdul es troba en aquesta situació, necessita la informació per determinar la posició relativa del cul de sac és i, d'aquesta forma, saber si ha de saltar o continuar endavant per evitar entrar-hi.

Hem realitzat proves per verificar les decisions que pren un mòdul amb tot tipus i formes de culs de sac. En l'exemple de la Figura 6.7 observem un cas de cantonada on el valor potencial milloraria en saltar. El mòdul actiu té un valor potencial (20) i l'estàtic un valor potencial (10). Per determinar si realment ha de saltar, el mòdul necessita conèixer el tipus de cul de sac al qual s'enfronta. En aquest cas es suficient amb la informació de les branques, ja que amb els valors mínim i màxim de cadascun sabem que les branques són diferents.

Un cop sabem que el valor potencial millora i que les branques són diferents, podem afirmar que, en aquest cas, si el mòdul continua endavant entrarà en un cul de sac, per tant, ha de saltar a l'altra branca.

En l'exemple també es pot veure com, en cas d'haver decidit continuar endavant, el mòdul s'hauria topat amb l'altre mòdul actiu que està sortint del cul de sac.



**Figura 6.7: Coll d'ampolla de cantonada de branques diferents amb millora de valor potencial.** Es mostren dues iteracions consecutives del salt (Concave Transition) del mòdul actiu (verd) d'una branca a una altra per evitar un cul de sac. La informació de la primera iteració dels dos mòduls rellevants mostra els seus valors potencials (comptador C00) i els valors de les seves branques (C03 per els valors mínims i C04 per els valors màxims).

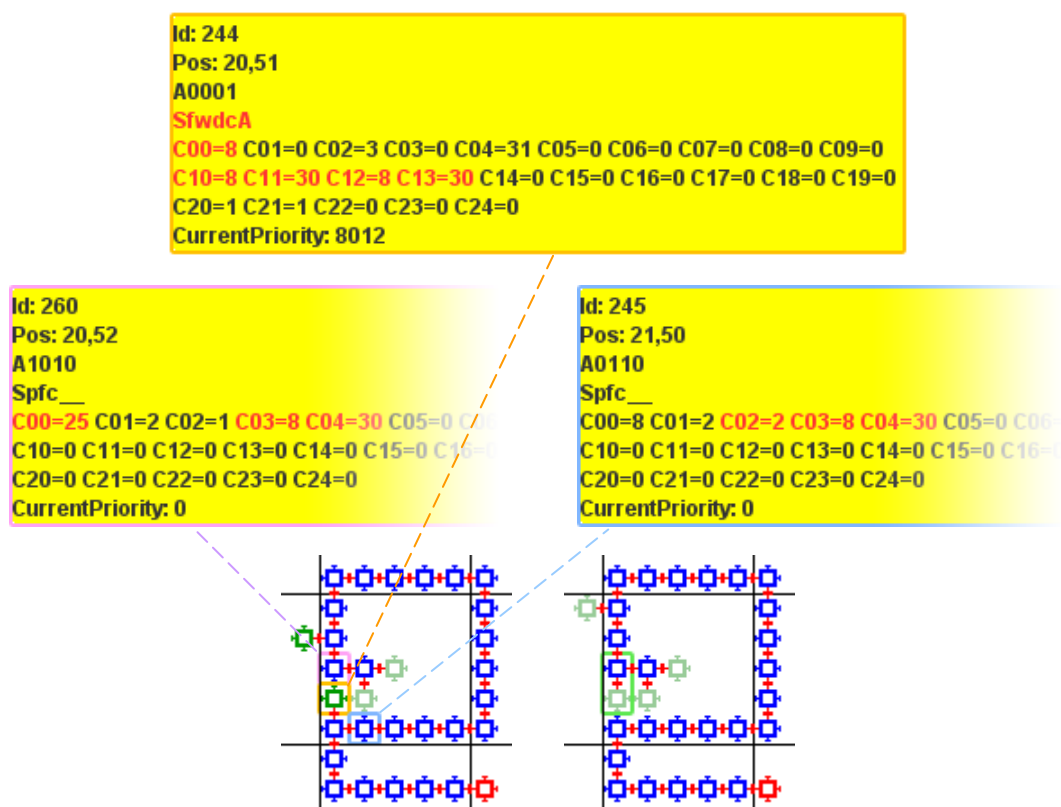
**6.7.3 Colls d'ampolla (cas especial):** en aquest cas ens trobem amb un altre tipus de cul de sac, un cas en el qual no es possible aconseguir la informació per identificar la forma del cul de sac sense realitzar una còpia prèvia de les dades necessàries.

Quan es dóna aquest cas, el procés de decisió triga dues iteracions. En la primera el mòdul agafa les dades que necessita i les guarda en comptadors propis. En la segona iteració compara aquestes dades per determinar el tipus de cul de sac de que es tracta. Un cop identificada la forma, el mòdul ja pot decidir si saltar o continuar.

En l'exemple de la Figura 6.8 observem que el dos mòduls estàtics (el marcat amb lila i el marcat amb blau) compleixen les condicions que determinen aquest cas especial. El mòdul del Nord té un valor potencial (25) pitjor que el mòdul actiu (8) i el mòdul del Sud-Est té el seu pare a l'Oest (2). Aquestes són les principals característiques per diferenciar el "cas especial" de la resta.

Un cop identificada la situació, el mòdul actiu copia les dades de les branques dels dos mòduls en 4 dels seus comptadors i canvia el seu estat per indicar que està decidint el camí. Al torn següent i en funció del resultat de la comparació, sap si ha de saltar o no.

En l'exemple, la branca dels dos mòduls estàtics és la mateixa i el valor potencial del mòdul actiu és pitjor que el del mòdul del Nord, per tant, en aquest cas ha de saltar per evitar el cul de sac.

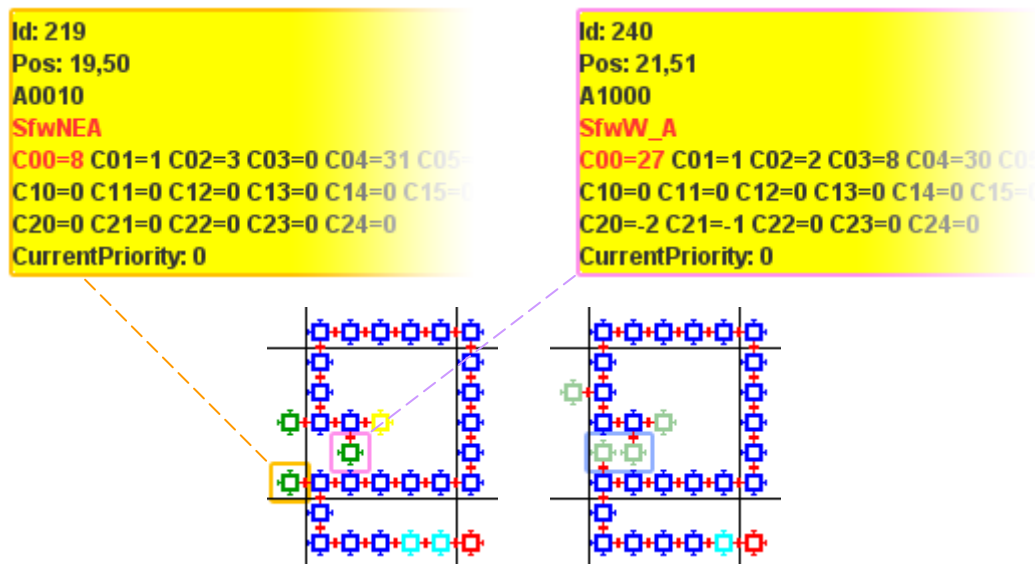


**Figura 6.8: Cas especial de cul de sac.** Es mostren dues iteracions consecutives del salt (Opposite Transition) del mòdul actiu (verd) d'una branca a una altra per evitar un cul de sac. La informació de la primera iteració dels tres mòduls implicats mostra les dades rellevants de cadascun. Del mòdul actiu: el seu estat, el valor potencial (C00), i els valors mínim i màxim del mòdul del Nord (C10 i C11) i els del mòdul del Sud-Est (C12 i C13). Del mòdul del Nord (lila): el valor potencial (C00) i els valors mínim i màxim de la branca (C03 i C04). I del mòdul del Sud-Est (blau): la posició del seu pare (C02) i els valors mínim i màxim de la branca (C03 i C04).

**6.7.4 Col·lisions (comunicació entre mòduls):** les col·lisions es produeixen quan dos mòduls volen ocupar una mateixa posició. Provem si amb el sistema de comunicació en dos torns implementat es pot evitar que dos mòduls col·lisionin.

En l'exemple de la Figura 6.9 ens trobem amb dos mòduls on un vol moure's al Nord-Est (NE) i l'altre a l'Oest (W\_). Per la posició en la qual es troben els mòduls, si tots dos apliquessin la seva intenció, xocarien.

Tal com estan implementades les regles, quan dos mòduls volen ocupar la mateixa posició té preferència aquell que té el valor potencial més petit. En aquest cas, el mòdul actiu (taronja) és qui guanya amb un valor potencial (8) i, per tant, es mou al seu Nord-Est tal i com havia anunciat. En canvi, el mòdul actiu (lila) amb un valor potencial (27) perd i es manté a l'espera.

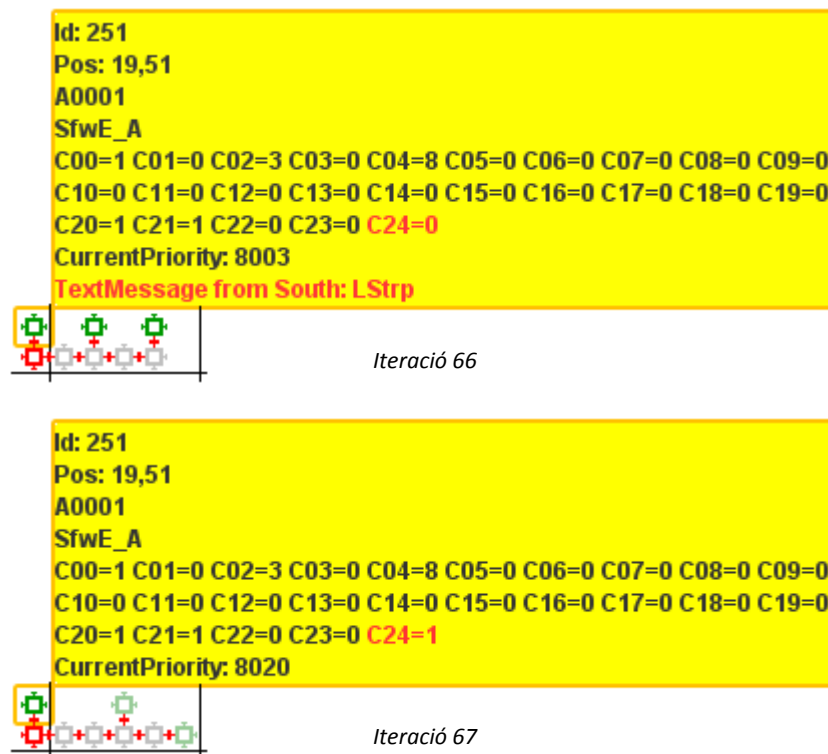


**Figura 6.9: Moviments amb possible col·lisió.** Es mostren dues iteracions consecutives que engloben el dos torns que triga en produir-se la comunicació entre mòduls en conflicte (comunicació d'intencions i moviment). Per cadascun d'aquest mòduls es mostra la informació rellevant que determina si es produeix un conflicte i, en cas de produir-se, quin dels dos mòduls aplica el moviment. La informació de la primera iteració dels dos mòduls implicats mostra els seus estats i els seus valors potencials (comptador C00).

**6.7.5 Marcatge de l'últim mòdul:** per finalitzar el procés necessitem un mòdul que determini el final de la realització de la fase. Hem decidit que aquest mòdul sigui l'últim dels actius.

En aquest cas, verifiquem que el màster és capaç d'identificar aquest mòdul per comunicar-li, i que el mòdul sap gestionar el missatge del màster per marcar-se.

En l'exemple de la Figura 6.10 observem com, en passar l'últim mòdul actiu sobre el màster, aquest li envia un missatge per indicar-li que ell és el màster. En la iteració següent, tot i que el mòdul actiu tenia la intenció de moure's a l'Est, abans de realitzar el moviment aplica la regla que marca el mòdul modificant un dels seus comptadors.



**Figura 6.10: Marcatge de l'últim mòdul actiu.** Es mostren dues iteracions consecutives en les quals s'observa la comunicació entre el màster i un mòdul actiu per indicar-li que és el líder de la tira. Es mostra la informació de l'últim mòdul actiu en les dues iteracions. En la iteració 66 rep el missatge del màster de "Leader Strip" (LStrp) i en la iteració 67 utilitza el comptador C24 per marcar-se.





## 7 Construcció de la forma final

### 7.1 Objectiu

L'objectiu d'aquesta fase és la de modificar el robot per crear una nova forma a partir de la forma canònica. Aquest procés es pot dividir en dues parts: aconseguir la informació de la forma final i moure els mòduls fins arribar a la seva nova posició.

La primera part consisteix en extreure de la forma final totes les dades que necessita cada mòdul per saber en quin moment ha arribat al seu destí i, un cop ha arribat, controlar el trànsit de la resta de mòduls.

La segona part és el moviment. Els mòduls han de recórrer el nou camí en ordre de tal manera que, amb la informació obtinguda, evitin les obstruccions i col·lisions. L'avenç dels mòduls es produeix tot seguint la regla de la mà esquerra.

La fase de construcció de la forma final acaba un cop tots els mòduls han arribat a la seva destinació.

*(El projecte finalitza a la primera part, tot i així, expliquem l'estudi que hem fet de l'estratègia a seguir a la segona part).*

### 7.2 Estratègia

A l'inici d'aquesta fase tots els mòduls formen part de la tirallonga o forma canònica.

La construcció de la forma final la dividim en dues parts: obtenció de la informació i construcció.

#### 7.2.1 Obtenció de la informació

En reconfiguracions no simulades, la informació de la forma final necessària per aquesta fase es podria posar en coneixement del màster des d'un controlador central. En el nostre simulador, això no és possible perquè cada mòdul està dotat de memòria i capacitat de computació constants. Així doncs, hem de dissenyar una estratègia diferent per tal que els mòduls coneguin la seva destinació final.

Amb aquest objectiu realitzem diverses tasques: generem el robot "mirall", donem valor potencial als mòduls del robot "mirall", obtenim les dades dels mòduls, i passem la informació als mòduls del robot original per utilitzar-la al llarg de la de construcció.

### 7.2.1.1 Robot “mirall”

El robot “mirall” és la configuració de la forma final representada en el simulador en un robot diferent de l’original, i en una posició diferent, que servirà com a font d’informació de les dades necessàries per la construcció de la forma final per part del robot original.

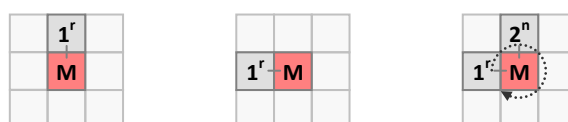
Per generar el robot “mirall”, fem el mateix procés de creació que amb el robot original. Un cop tenim els mòduls que formen la nova configuració del robot posicionats i enganxats, realitzem un procés similar al realitzat amb la configuració inicial del robot original.

### 7.2.1.2 Valor dels mòduls “mirall”

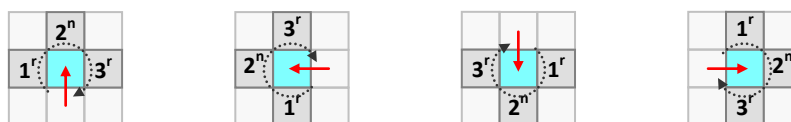
Per poder donar valor potencial als mòduls realitzem la cerca del màster, la detecció de forats, la construcció de l’scan tree i la construcció de la funció potencial amb la nova configuració.

La **cerca del màster, dels líders de forat i la construcció de l’arbre** segueixen exactament el mateix mètode que fem servir amb la forma inicial (vegeu els capítols 2, 3 i 4). Un cop hem identificat el màster i els líders de forat i hem construït l’arbre, passem a assignar els valors potencials dels mòduls, així com els valors mínims i màxims identificatius de les branques.

En aquest cas, la **fase de construcció de la funció potencial**, és la mateixa però seguint l’ordre invers (vegeu el Capítol 5). Per construir la funció potencial de la forma final utilitzem un recorregut en profunditat per l’arbre però, en aquest cas, amb prioritat horària. A la Figura 7.1 es pot veure l’ordre d’inici des del màster i a la Figura 7.2 l’ordre de propagació dels missatges.



**Figura 7.1: Propagació del missatge d'un mòdul màster (reverse).** Es mostra per quin mòdul es propaga el missatge en la fase de construcció de la forma final. En cas de tenir més d'una possibilitat la preferència és per l'Oest.



**Figura 7.2: Propagació del missatge d'un mòdul (reverse).** Es mostra l'ordre que segueixen les regles per decidir per on propagar el missatge en la fase de construcció de la forma final. L'ordre ve determinat en funció de per on ha arribat el missatge (fletxa vermella). L'ordre és en sentit horari (fletxa negra).

De la mateixa manera que amb la forma inicial, el procés finalitza un cop arriba el missatge de tornada al màster i no queden més branques per les quals propagar aquest missatge.

### 7.2.1.3 Obtenció de dades “mirall”

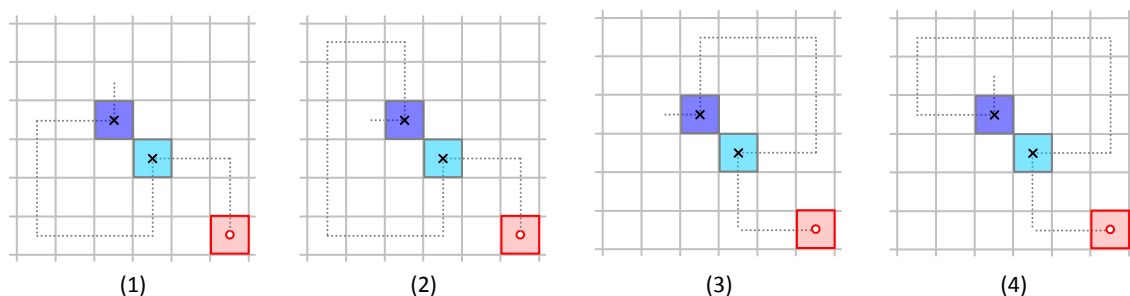
Un cop els mòduls han completat la construcció de la funció potencial tenim les dades bàsiques (valor potencial, pare, màxim i mínim) però els mòduls que formen part d'un coll d'ampolla necessiten dades extra per tal de distribuir la resta mòduls en moviment i evitar que entrin en el cul de sac si no formen part del seu interior, i així evitar les obstruccions.

Un coll d'ampolla el formen dos mòduls, un amb valor potencial més petit (està més a prop del màster) al qual anomenarem **líder del sac** i un altre amb valor potencial més gran que anomenarem **veí del sac**. El mòdul líder del sac és qui ha d'agafar les dades del veí del sac. Tots els mòduls es mouen d'un valor potencial més petit (el màster) fins un més gran (el seu propi valor potencial), per tant, el veí del sac no necessita informació perquè cap mòdul saltarà a un mòdul amb valor potencial més petit.

El líder del sac agafa dos tipus de dades de l'altre mòdul: el valor potencial i el valor màxim. Tal com està generat l'arbre i de la forma que hem assignat els valors potencials, depenent de la forma del cul de sac, aquestes dades serveixen per determinar quins mòduls entren al cul de sac i quins mòduls salten. Això té per objectiu garantir que els mòduls que salten no alterin l'ordre de l'avenç (vegeu més detalls a l'Apartat 7.2.2.4).

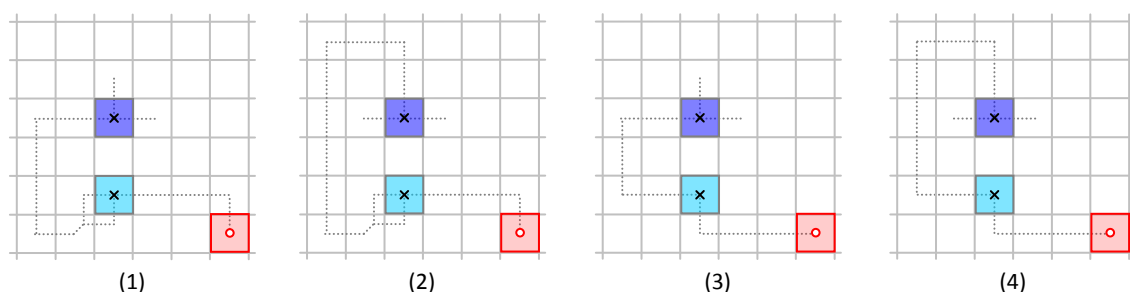
Hi ha diverses formes de cul de sac en funció dels dos mòduls que formen part del coll d'ampolla i dels seus veïns:

- **Dues subbranques diferents:** si cadascun dels mòduls del coll d'ampolla pertany a una sub branca diferent, en cap cas necessitem capturar dades. La raó és que primer es forma una de les subbranques i després l'altra. Cap dels mòduls d'una sub branca circula per l'altra, per tant, cap mòdul ha de saltar entre subbranques.
- **Mateixa branca (cantonada):** a la Figura 7.3 podem veure tots els casos de cantonada. Només en el primer cas necessitem agafar les dades per determinar els mòduls que salten del líder del sac al veí del sac, ja que la seva destinació es troba més enllà d'aquest. A la resta de casos o bé els mòduls circulen per l'exterior del cul de sac, o bé els mòduls formen part de l'interior del cul de sac i, en tot cas, cap no necessita saltar del líder del seu veí.

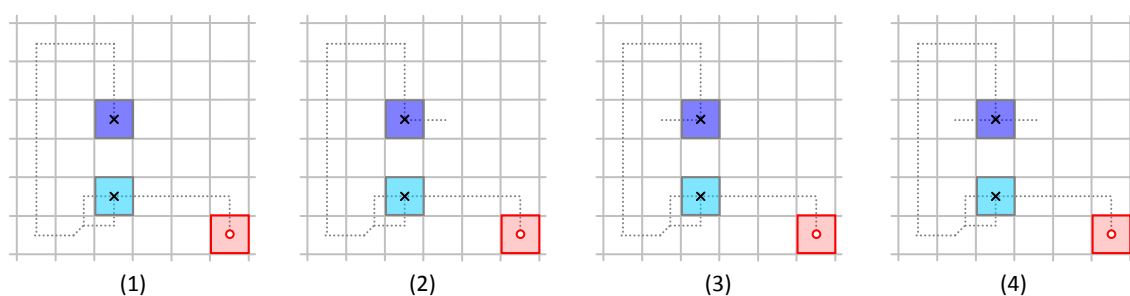


**Figura 7.3: Formes de cul de sac (coll d'ampolla en cantonada).** Es mostren els casos de cul de sac que es poden donar si els mòduls que causen el coll d'ampolla estan en cantonada. El mòdul blau cel és el líder del sac, el blau fosc és el veí del sac i el vermell on es troba el màster.

• **Mateixa branca (front):** a la Figura 7.4 podem veure els casos conflictius, a la resta de casos els mòduls circulen per l'exterior del cul de sac o formen part del seu interior. En els casos 1 i 3 necessitem capturar les dades en totes les seves variants. Les variants de cada cas són les diferents direccions per on pot continuar la branca del líder del sac i la branca o branques que surten del veí del sac. En els casos 2 i 4 la captura de les dades depèn de les branques que surten del veí del sac. Desglossem el cas 2 en 4 variants a la Figura 7.5 (amb el cas 4 succeeix el mateix). A la variant 1, cap mòdul no ha de saltar. A la variant 2 agafem les dades per saber quins mòduls han de saltar (els de la subbranca dreta del veí del sac). A la variant 3, pel fet que els mòduls de la branca pertanyen a l'interior del cul de sac, no necessitem cap dada ja que cap mòdul no ha de saltar. Per últim, a la variant 4 també necessitem saber quins mòduls salten, però en aquest cas, en ser una bifurcació de dues subbranques, el valor màxim l'hem d'agafar de la subbranca on es troben els mòduls que han de saltar (la subbranca exterior al cul de sac), aquest cas de cul de sac és el que anomenem "especial".



**Figura 7.4: Formes de cul de sac (coll d'ampolla de front).** Es mostren els casos d'orientació Nord de cul de sac que es poden donar si els mòduls que causen el coll d'ampolla estan de front. El mòdul blau cel és el líder del sac, el blau fosc és el veí del sac i el vermell on es troba el màster.



**Figura 7.5: Variants del cul de sac (amb bifurcació) de la Figura 7.4 (2).** Es mostren les variants que es poden donar quan el veí del sac és una bifurcació. (1) Sense branques. (2) Branca externa. (3) Branca interna. (4) Bifurcació, dues subbranques. El mòdul blau cel és el líder del sac, el blau fosc és el veí del sac i el vermell on es troba el màster.

Cada mòdul té 8 direccions potencials (N, S, E, W, NE, NW, SE, SW) on pot haver-hi un coll d'ampolla, per tant, ha de poder emmagatzemar 16 dades (dues per cada direcció). I els líders de sac s'encarreguen de distribuir la circulació dels mòduls per cadascuna d'aquestes direccions.

Per altra banda, hem d'assignar **responsables de cada forat** per tal d'evitar el seu tancament mentre algun mòdul extern al forat continuï circulant pel seu interior o a l'inrevés. Aquest responsable és l'últim mòdul pel qual passa el líder de forat o el veí del líder abans d'arribar al seu destí. El responsable agafa el valor potencial del mòdul que realitza el tancament del forat (vegeu dos exemples a la Figura 7.6).



**Figura 7.6: Responsables de forat.** Es mostren dos casos diferents de forat. El mòdul taronja és el líder de forat, el groc el veí del líder, el blau el responsable del forat. En aquests casos el responsable (blau) agafa el valor potencial del veí del líder de forat (groc) i controla que el forat no es tanqui abans d'hora.

#### 7.2.1.4 Utilització de les dades “mirall”

L'obtenció de la informació per part dels mòduls del robot original es realitza quan el mòdul arriba a la seva destinació. En arribar-hi agafa totes les dades del mòdul “mirall” i les emmagatzema. En cas de ser un líder de sac o un responsable de forat, passa a realitzar la funció de controlador (vegeu l'Apartat 7.2.2.4).

### 7.2.2 Construcció

Un cop els mòduls han completat la fase de reconfiguració a la forma canònica i tots els mòduls formen part de la tirallonga, els mòduls poden començar a tornar cap el màster i iniciar el camí a la nova configuració, la forma final. La informació que proporcionen els mòduls del robot “mirall” ha d'estar disponible abans de començar aquesta fase.

La construcció de la forma final es pot separar en quatre parts: inici de la construcció, obtenció de les dades de destinació, moviment fins la posició final i finalització de la construcció.

#### 7.2.2.1 Inici de la construcció

L'inici de la construcció el determina el mòdul, identificat com a “últim mòdul actiu” en la fase anterior, un cop adherit a la tira. Aquest mòdul es torna a activar i inicia el procés de construcció. En reactivar-se un mòdul, comunica al seu veí (de l'Oest) que pot activar-se tan bon punt sigui possible. La condició per tal que un mòdul pugui activar-se és que no ha d'estar connectat a cap altre mòdul actiu. Un cop activat, el mòdul es mou per la tira fins arribar al màster.

Tot moviment d'aquesta fase és invers al de la fase prèvia (reconfiguració a la forma canònica). En aquesta fase, el moviment segueix dues pautes: la **regla de la mà esquerra** i el **moviment sobre mòduls estàtics**. La primera serveix per determinar la direcció en la qual circula un mòdul i la segona evita que un mòdul quedi desconnectat del robot.

#### **7.2.2.2 Dades de destinació**

L'obtenció de les dades de destinació es realitza quan el mòdul passa pel màster. El màster proporciona als mòduls les coordenades  $x$  i  $y$  de referència i el valor potencial final del mòdul quan arribi a la seva destinació. Les coordenades  $x$  i  $y$  indiquen la posició on es troba el màster del robot "mirall", i el valor potencial indica quin mòdul és dels que formen part de la configuració final. El valor potencial és 1 pel primer mòdul i es va incrementant amb cada mòdul que passa pel màster.

#### **7.2.2.3 Moviment del robot original**

El moviment dels mòduls fins la posició final consisteix en moure's (seguint les dues pautes esmentades) fins arribar a la seva posició de destinació. En cada moviment, el mòdul verifica si el mòdul que es troba en la posició anàloga de la configuració "mirall" (és a dir, a la posició relativa  $x, y$ ) coincideix en valor potencial. Si la resposta és afirmativa vol dir que ha arribat al seu destí sinó, el mòdul ha de continuar movent-se.

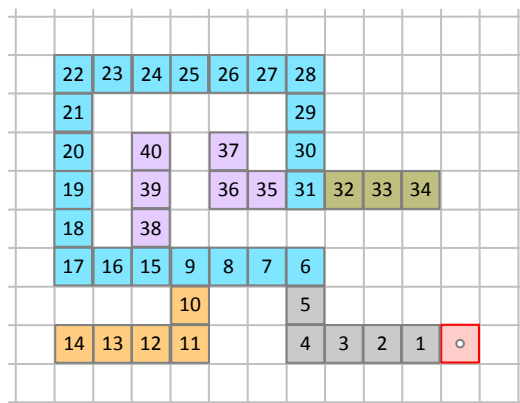
#### **7.2.2.4 Control i final de la construcció**

Quan un mòdul arriba a la seva posició final, agafa totes les dades del seu mòdul "mirall" i, depenent de les dades obtingudes, passa a mode controlador. Els mòduls que han de realitzar la funció de controlador són els líders de sac i els responsables de forat. La resta, tot simplement, es desactiven.

El **líder del sac** ha de decidir quins mòduls continuen i quins mòduls salten (o s'esperen en cas que el veí del sac encara no hagi arribat a la seva posició).

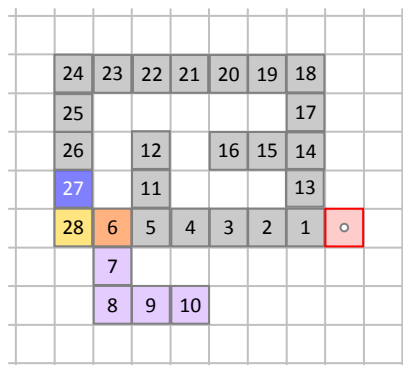
Si un mòdul no té cap dada deixa circular a tots els mòduls. Si el valor potencial del mòdul que prova de passar és més petit que el del valor potencial capturat (el del veí del sac), llavors el deixa passar perquè forma part d'una branca anterior o de la branca del cul de sac. Si el valor potencial és més petit que el valor màxim capturat (i més gran que el valor potencial) llavors el líder del sac diu que ha de saltar al mòdul actiu perquè forma part de branques externes al cul de sac. Si el valor potencial del mòdul és més gran que el valor màxim capturat, llavors el deixa passar perquè forma part de branques internes del cul de sac.

A l'exemple de la Figura 7.7 observem la distribució que fa el líder de sac amb els mòduls del robot "mirall". El líder del sac és el mòdul amb valor potencial (6) i les dades capturades són el valor potencial (31) i el màxim (35), per tant, deixa passar els mòduls fins el (31), salten del (32) al (34) i entren del (35) en endavant.



**Figura 7.7: Exemple de distribució dels mòduls.** Es mostra un exemple de cul de sac "especial" amb els mòduls numerats amb el seu valor potencial (que coincideix amb l'ordre pel qual passen pel màster). El mòdul (6) és el líder del sac i el (31) el veí del sac. Els mòduls blaus pertanyen a la branca del cul de sac. Els mòduls taronges pertanyen a una branca anterior al cul de sac. Els mòduls verds pertanyen a una branca exterior/posterior al cul de sac. Els mòduls liles pertanyen a branques interiors al cul de sac.

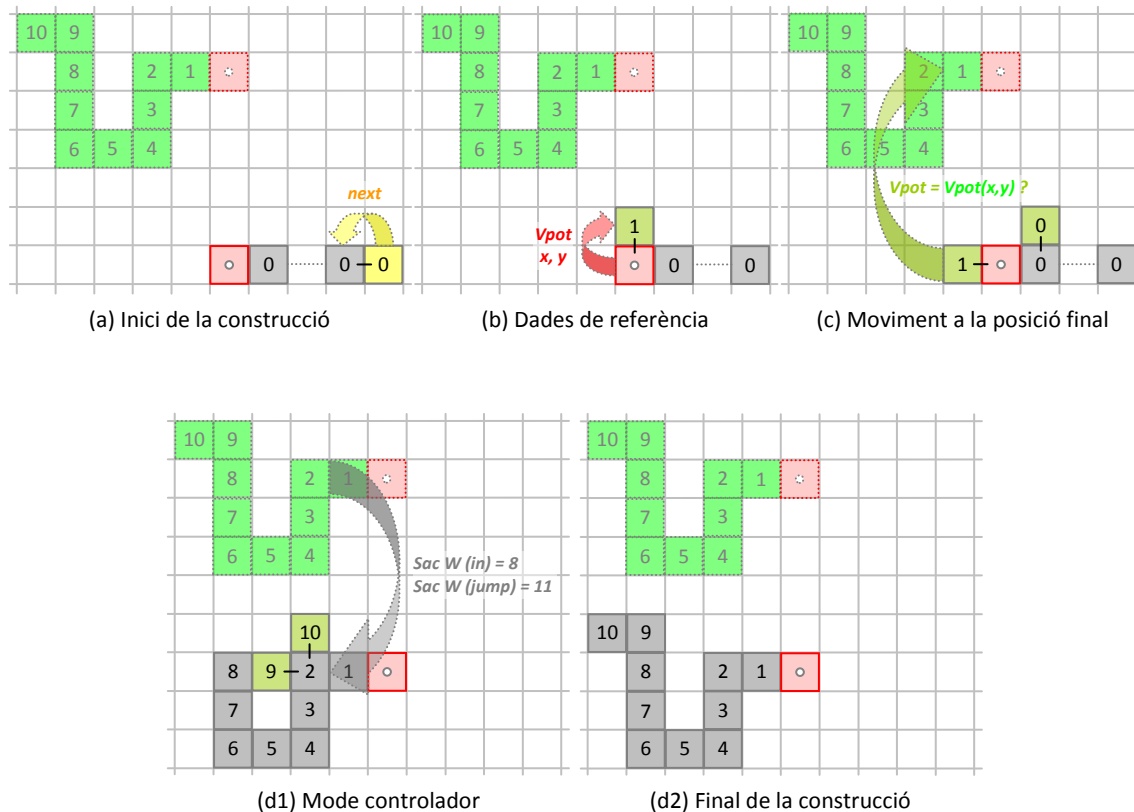
El **responsable de forat** ha de frenar el mòdul que té al seu càrrec per evitar que tanqui el forat abans d'hora. El mòdul que tanca el forat es manté a l'espera fins que tots els mòduls que es troben dintre del forat i que no formen part del seu interior surten a l'exterior (vegeu un exemple a la Figura 7.8).



**Figura 7.8: Exemple de responsable de forat.** Es mostra un exemple d'una configuració amb forat amb els mòduls numerats amb el seu valor potencial. El mòdul taronja és el líder de forat, el groc el veí del líder, el blau el responsable del forat i els liles els mòduls externs al forat. En aquest cas el responsable (27) ha d'aturar al mòdul (28) fins que els mòduls (7, 8, 9 i 10) hagin sortit del forat (en cas que encara no ho hagin fet).

Aquest és el cas general. Hi ha un tipus de forat "especial" on el mòdul responsable, a diferència del que passa amb un forat normal, ha de permetre que entrin els mòduls que pertanyen al seu interior. Aquest tipus de forats necessiten un tractament especial. L'estratègia utilitzada amb els forats "especials" es troba detallada a l'Apartat 7.4.2.

A la Figura 7.9 podem veure un exemple amb les quatre parts del procés. En (a) s'activa el primer mòdul, avisa el següent i inicia el moviment en direcció al màster. En (b) el màster comunica al mòdul actiu les dades de referència del robot "mirall". En (c) el mòdul actiu (1) comprova si es troba a la seva posició final mirant si el seu valor potencial correspon al del mòdul que es troba a les coordenades de referència. En (d) el mòdul (2) manté els mòduls actius (9) i (10) a l'espera fins que poden "saltar" al mòdul (8) evitant el cul de sac i completant la construcció.



**Figura 7.9: Procés de construcció.** Es mostra el procés que segueix el robot original per passar de la forma canònica a la forma final. La configuració superior (de verd) representa la forma final, els mòduls "mirall" que contenen la informació de la configuració final. El robot inferior és el robot original, el que realitza la reconfiguració.

## 7.3 Regles

Les regles descrites a continuació són les encarregades de generar les dades de la forma final que necessita el sistema per tal que els mòduls puguin construir la forma final (la part d'obtenció de la informació). Les regles que realitzen la construcció de la forma final pròpiament dita no estan implementades (la part de construcció).



Les regles per dur a terme la part d'obtenció de la informació estan subdividides en 3 grups, *reverse*, *rollback*, i obtenció de la informació. El primer grup inclou totes les regles que realitzen la cerca del màster, la detecció dels forats, la construcció de l'arbre i la construcció de la funció potencial de la configuració "mirall"; el segon grup s'encarrega de detectar formes de forat "especials" i desfer la construcció de la funció potencial feta; i el tercer grup és qui s'encarrega de detectar els líders dels sacs i els responsables dels forats per aconseguir les seves dades.

Les regles d'aquesta fase s'identifiquen per la inclusió de "[R]" (*Reverse*) a l'inici del nom de les regles, i la modificació del primer caràcter de tots els estats pel caràcter 'R'. Aquesta 'R' serveix per diferenciar les regles que afecten a la forma "mirall". Aquesta diferenciació permet tractar alhora la forma original i la forma "mirall" en el simulador.

La resta de caràcters de l'estat segueixen les mateixes pautes que a la resta de regles. Per tant, la identificació dels mòduls "mirall" és la següent:

[RNDEF]	Estat inicial. <i>Reverse</i> .
[RniM0]	Candidat a màster ( <i>Master</i> ). <i>Reverse</i> .
[RniCM]	Missatge de candidat a màster YES ( <i>Candidate Master</i> ). <i>Reverse</i> .
[RniCS]	Missatge de candidat a màster NO ( <i>Candidate Slave</i> ). <i>Reverse</i> .
[RniH0]	Candidat a líder de forat ( <i>Hole</i> ). <i>Reverse</i> .
[RniCH]	Missatge de candidat a líder de forat YES ( <i>Candidate Hole</i> ). <i>Reverse</i> .
[RniCL]	Missatge de candidat a líder de forat NO ( <i>Candidate Line</i> ). <i>Reverse</i> .
[Rst_M]	Màster en la fase build scan tree – <i>Reverse</i> .
[Rst_H]	Líder de forat en la fase build scan tree – <i>Reverse</i> .
[Rst_]	Resta de mòduls en la fase build scan tree – <i>Reverse</i> .
[Rpf*M]	Màster en la fase build potential function – <i>Reverse</i> .
[Rpf*H]	Líder de forat en la fase build potential function – <i>Reverse</i> .
[Rpf_]	Mòduls en espera del missatge de valor potencial. <i>Reverse</i> .
[Rpfm]	Mòduls fent el recorregut de marcatge (anada). <i>Reverse</i> .
[Rpfw]	Mòduls fent el recorregut de marcatge (en espera). <i>Reverse</i> .
[Rfc_]	Mòduls fent el recorregut de marcatge (tornada). <i>Reverse</i> .
[Rgi_M]	Màster en la fase build final shape – <i>Reverse</i> (informació disponible).
[Rgi_H]	Líder de forat en la fase bfs – <i>Reverse</i> (informació disponible).
[Rgi_]	Resta de mòduls en la fase bfs – <i>Reverse</i> (informació disponible).

Per a una descripció detallada de les regles, vegeu l'Annex D.

### **7.3.1 Grup: “reverse rules”**

Aquest grup està format per les mateixes regles que els seus grups anàlegs de la versió normal (no invertida) dels capítols 2, 3, 4 i 5. Les regles de les fases de cerca del màster, detecció dels forats, construcció de l'*scan tree* i construcció de la funció potencial són les que s'inclouen en aquest grup.

Aquestes regles segueixen el mateix procés però invertint l'ordre de prioritat a l'hora de decidir la direcció, seguint així la regla de la mà esquerra. L'estructura resultat és la mateixa que en utilitzar les regles “normals” (mateix màster, mateixos forats i mateix arbre) però amb diferent valor potencial.

### **7.3.2 Grup: “rollback”**

Aquest grup està format per 4 regles: 1 regla que inicia el *rollback* i 3 que el propaguen.

El *rollback* es realitza en trobar un forat “especial” del qual hem de modificar la seva estructura (vegeu l'Apartat 7.4.2). Aquests tipus de forats es localitzen al llarg de la fase de construcció de la funció potencial (*reverse*) en detectar que el veí del líder de forat té un valor potencial més petit que el del líder. En canviar l'estructura, les dades generades per la aquesta fase deixen de ser correctes, per tant, realitzem un *reset* de tota aquesta informació i tornem a iniciar la fase des del màster.

El *rollback* consisteix en un missatge de *reset* que s'estén en amplada per tots els mòduls connectats. Quan el missatge de *reset* arriba al màster, inicia de nou el procés de construcció de la funció potencial. Si el màster té dues branques i la modificació de l'estructura només afecta una d'elles, pot iniciar la fase a partir de la branca afectada.

### **7.3.3 Grup: “get info”**

Aquest grup està format per 15 regles subdividides en 3 subgrups. El primer subgrup consta de 12 regles per capturar les dades dels mòduls que generen els coll d'ampolla, 8 regles per als mòduls que es troben de front (4 per bifurcacions normals, 4 per bifurcacions especials) i 4 per els mòduls que estan en cantonada. El segon subgrup té 2 regles, una per cada tipus de responsable de forat depenent de la forma que té el tancament del forat. El tercer subgrup és d'una única regla que canvia l'estat del mòdul si ha capturat tota la informació possible del seu entorn.

Les bifurcacions normals són aquelles en les quals el veí del sac té una prolongació de la seva branca per l'exterior del cul de sac. En canvi, les bifurcacions especials són aquelles on el veí del sac té dues bifurcacions, una que va per l'exterior del cul de sac i una altra que va per l'interior.

Per detectar un coll d'ampolla un mòdul ha de mirar els buits del seu voltant. Un cop identificat si el mòdul és el líder del sac (té valor potencial més petit que l'altre mòdul involucrat), obté les dades del veí del sac per conèixer quins mòduls haurà de deixar passar i quins haurà de fer saltar. Un mòdul pot ser líder de més d'un sac i el sac es pot donar en qualsevol direcció, per aquesta raó reservem comptadors per cada possible direcció.

Nota: En les regles dels responsables de forat, es contemplen dos tipus de forats en funció de la seva forma. Amb la modificació de l'estructura causada pels forats "especials" es poden produir nous tipus (*Get special hole responsible*) que estan pendents d'anàlisi per si requereixen un tractament específic.

La regla que s'encarrega de canviar l'estat del mòdul i donar per finalitzat el procés de captura de dades verifica l'estat de tots els mòduls que poden formar part d'un coll d'ampolla. Si tots aquests mòduls de l'entorn es troben en una fase en la qual disposen de tota la informació per extreure, vol dir que el mòdul ha agafat tota la informació possible en torns anteriors i, per tant, pot assegurar que disposa de totes les dades que necessita el robot original. La prioritat més petita d'aquesta regla fa que no pugui aplicar fins que el mòdul no ha capturat l'última dada.

## 7.4 Problemes

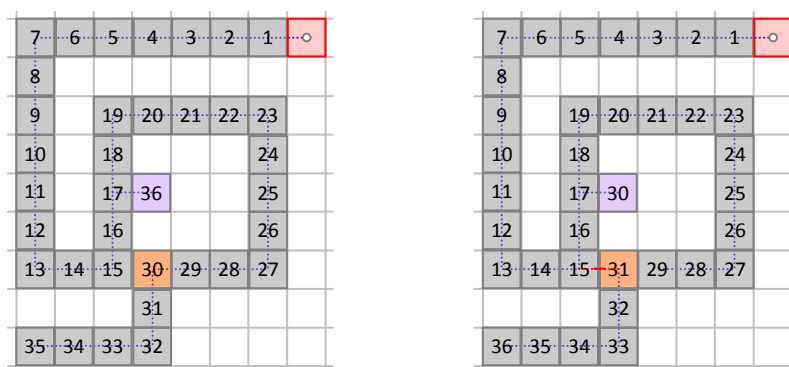
Pel fet que la part de "reverse" és similar a les fases de capítols anteriors, molts dels problemes que podrien haver sorgit ja han estat solucionats prèviament. Aquests problemes no els tornarem a comentar en aquest apartat. Expliquem els problemes que han sorgit referent a les parts de *rollback* i captura de dades i quina solució hem adoptat, i els que creiem que poden sorgir de la part que no s'ha implementat i la solució que hem pensat.

**7.4.1 Forats:** hi ha forats que poden tancar-se deixant mòduls que no pertanyen a l'interior del forat tancats a dintre i sense possibilitats de sortir. Aquests forats estan formats per dues branques que, durant la construcció de la forma final, finalitzen la construcció tancant el forat abans que els mòduls que encara estan circulant per l'interior tinguin temps de sortir.

*Solució adoptada: assignem a un mòdul la tasca de controlar al mòdul que tanca el forat. El responsable de forat ha d'aturar al mòdul que tanca fins que tots els mòduls que no pertanyen a l'interior del forat hagin sortit.*

**7.4.2 Forat “especial”:** hi ha un tipus de forat (d’una branca) en el qual pot succeir que durant la construcció es generi el forat i es tanqui de manera que els mòduls que formen part de branques interiors del forat ja no poden entrar.

*Solució adoptada:* modifiquem l’estructura de l’arbre perquè el forat passi a tenir dues branques i evitar així que mòduls interns tinguin un valor potencial més gran que l’estructura del forat. En l’exemple de la Figura 7.10 podeu veure com, abans de modificar la connexió del líder de forat, el mòdul lila es quedaria fora a causa de tenir un valor potencial més gran i arribar a la seva posició un cop el forat ja està tancat, en canvi, després de modificar l’estructura el mòdul lila arribaria al seu destí abans no es produeixi el tancament del forat.



**Figura 7.10: Forat “especial”.** Es mostra un exemple d’una configuració amb forat amb els mòduls numerats amb el seu valor potencial. La configuració de l’esquerra és prèvia a la modificació, la de la dreta és després del canvi de connexió del líder de forat. El mòdul taronja és el líder de forat i el lila un mòdul de l’interior del forat.

**7.4.3 Cul de sac “especial”:** hi ha un tipus de cul de sac en el qual el màxim que proporciona el veí del sac no coincideix amb els mòduls que han de saltar. Aquest cas es dóna quan el veí del sac té dues bifurcacions i una va a l’interior del cul de sac, i l’altra va a l’exterior.

*Solució adoptada:* implementem regles específiques en les quals, si es dóna aquest cas, el líder del sac captura el valor màxim (que determina quins mòduls han de saltar) d’un mòdul de la branca externa en comptes d’agafar-ho del veí del sac.

**7.4.4 Col·lisions:** quan entre dues branques diferents hi ha espai per un únic mòdul, depenent de la direcció de les branques, es poden produir col·lisions entre dos mòduls que circulen en direccions oposades, cadascun per una branca.

*Solució proposada:* hauríem de fer que els mòduls en conflicte es canviïn els papers. Els dos mòduls afectats intercanvien les seves connexions, els seus estats i les dades dels seus comptadors de manera que cada mòdul realitza el paper de l’altre.

**7.4.5 Obstruccions:** de la mateixa manera que amb les col·lisions, si els mòduls involucrats queden un al costat de l'altre en direccions oposades, en aquest cas es produeix una obstrucció.

*Solució proposada: la solució que proposem és la mateixa que en el cas de les col·lisions, intercanviar els papers.*

**7.4.6 Informació del màster:** a causa de les característiques del màster, pot tenir una branca pel Nord que impedeix que mòduls en procés de construcció de la forma final passin per sobre per agafar les dades de referència.

*Solució proposada: canviar les propietats del màster o fer que la informació la proporcioni un altre mòdul que passi per tots els mòduls comunicant la informació (per exemple el "últim mòdul actiu", l'últim de la tirallonga).*

Durant el transcurs d'aquesta fase no s'han produït gaires incidències pel fet que no hem completat la fase i, per ara, les regles implementades serveixen per realitzar les tasques plantejades. La complexitat en aquest cas ha estat més la quantitat de regles necessàries que la quantitat de problemes ocasionats.

## 7.5 Modificacions

No hem realitzat cap modificació. Donat que el procés de construcció de la forma final no s'ha completat, no sabem si serà necessària alguna modificació en la implementació que hem fet o alguna ampliació amb noves dades que siguin necessàries per realitzar la part de la construcció.

## 7.6 Alternatives

En aquesta fase hi ha tres qüestions que s'havien plantejat de forma diferent. Una de caire global per realitzar la tornada i dues per tasques concretes: la captura de dades i el *rollback*.

En un principi, la **construcció de la forma final** no estava plantejada de la forma que finalment hem implementat. La idea inicial era obtenir les coordenades de destí de cada mòdul i l'instant de temps en el qual passen pel màster. D'aquesta forma, en passar pel màster en el procés de construcció de la forma final, el màster controla en quin moment el mòdul ha d'iniciar el procés. La idea és que si els mòduls tornen en l'ordre invers amb els mateixos intervals de temps, la construcció es pot dur a terme com si els mòduls fessin marxa enrere. Aquest mètode es descarta pel fet que no podem controlar els salts. Al llarg del procés de reconfiguració a la forma canònica, els mòduls poden saltar avançant-se a uns altres i trencant totalment l'ordre.

La **captura de dades** l'hem fet aprofitant les característiques del simulador i generant les dades que necessitem aplicant regles a un nou robot (que anomenem "mirall") en el qual es basen els mòduls del nostre robot original per obtenir les dades que necessiten per construir la forma final. Aquestes dades no tenen perquè generar-se en el propi simulador, pot haver-hi un procés extern que proporcioni tota aquesta informació al màster, i el màster sigui l'encarregat de distribuir aquesta informació als mòduls que realitzen el procés de construcció de la forma final.

Per solucionar la incongruència de les dades en realitzar un canvi en l'estructura de l'arbre per causa d'un forat "especial", decidim fer un **rollback** per tornar a l'estat previ a l'assignació d'aquestes dades. El **rollback** que apliquem arriba fins el màster i afecta a tots els mòduls de la branca, però la quantitat de mòduls afectats es pot reduir. El **rollback** es pot fer de manera que arribi només fins el veí del líder de forat (que és on es produeix la modificació de les dades). El problema radica en el fet que el missatge que transportava la informació de la construcció de la funció potencial s'ha perdut, per tant, seria necessari implementar regles que puguin recuperar aquesta informació i reactivar el missatge.

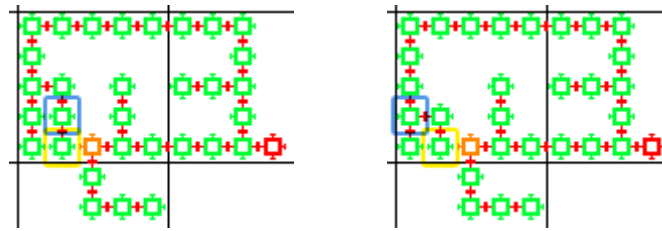
## **7.7 Models de prova**

Per verificar el correcte funcionament d'aquesta fase, hem realitzat diferents proves de cada part del procés. Aquestes proves inclouen la revisió de les "*reverse rules*" però, degut a la seva similitud amb les realitzades en capítols anteriors, en aquest apartat només tractem casos conflictius que no s'han produït en fases prèvies.

**7.7.1 Responsable de forat:** per tal d'evitar el tancament d'un forat abans que surtin els mòduls que no pertanyen a l'interior del forat, necessitem que un mòdul determinat prengui la responsabilitat d'aturar al mòdul que produeix aquest tancament.

A la Figura 7.11 observem dos casos on, depenent de l'estructura de l'arbre, el mòdul que ha de prendre la responsabilitat del mantenir el forat obert varia.

Les regles implementades diferencien aquests casos i atorguen la responsabilitat al mòdul adient. Si en el cas de la dreta el responsable del forat fos el mateix que en el de l'esquerra, el responsable aturaria el mòdul un cop ja ha tancat el forat. En ambdós casos, el responsable de forat (mòdul marcat de blau) té el valor potencial del mòdul que ha d'aturar (mòdul marcat de groc) en el comptador C01. La resta de mòduls no responsables tenen el valor zero en el comptador.



**Figura 7.11: Responsables de forat.** Es mostren dos casos similars on el responsable de forat es troba en posicions diferents. El mòdul taronja és el líder de forat, el mòdul marcat de groc és el veí del líder, el mòdul marcat de blau és el responsable de forat.

**7.7.2 Forat “especial”:** el forat especial és aquell que per la seva forma ha de modificar l'estructura de l'arbre per evitar que mòduls amb valor potencial més gran pertanyents a l'interior del forat es quedin fora sense poder-hi entrar.

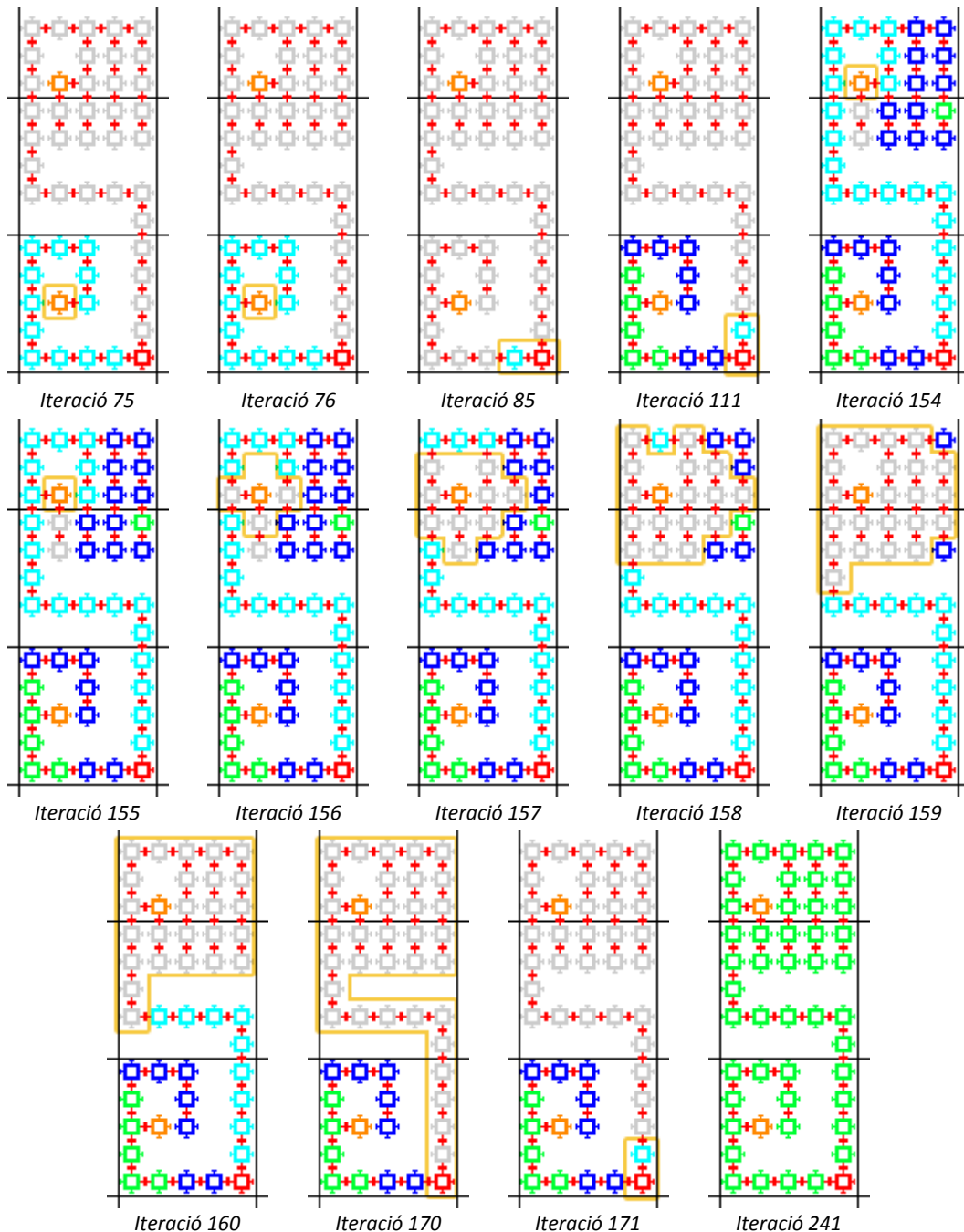
Les proves realitzades mostren que les regles implementades detecten aquest tipus de forat i modifiquen l'estructura de l'arbre intercanviant la connexió Est i Oest del mòdul “líder de forat”. Aquest canvi produeix un missatge de *rollback* que tractem a l'Apartat 7.7.3.

A la Figura 7.12 podem observar com a la iteració 75 es detecta el forat “especial” modificant la connexió del líder de forat a la iteració 76, el mateix succeeix en les iteracions 154 i 155.

**7.7.3 Rollback:** a causa de la localització d'un forat “especial”, es modifica l'estructura de l'arbre provocant que les dades que s'han anat assignant als mòduls ja no tinguin validesa. Per reassignar aquesta informació, netegem les dades de tots els mòduls, i iniciem de nou (des del màster) el procés de construcció de la funció potencial.

A l'exemple de la Figura 7.12 mostrem un cas amb un forat “especial” per branca. A la iteració 76 s'ha modificat l'estructura de l'arbre, i això provoca un *rollback* que arriba fins el màster. El màster inicia de nou el procés de construcció de la funció potencial a la iteració 85. A la iteració 155 es modifica de nou l'estructura de l'arbre i en les successives iteracions, de la iteració 156 a la 160, es pot veure l'expansió del missatge de *reset*.

A la iteració 170 el missatge de *reset* arriba al màster però, a la iteració 171 el màster és capaç de reprendre el procés de construcció de la funció potencial només per la branca afectada sense necessitat de resetejar l'altra branca. Per fer això, el màster necessita agafar la informació del mòdul de la branca no afectada. Finalment, a la iteració 241, les dades de cada mòdul han estat assignades satisfactòriament amb l'estructura de l'arbre modificat.



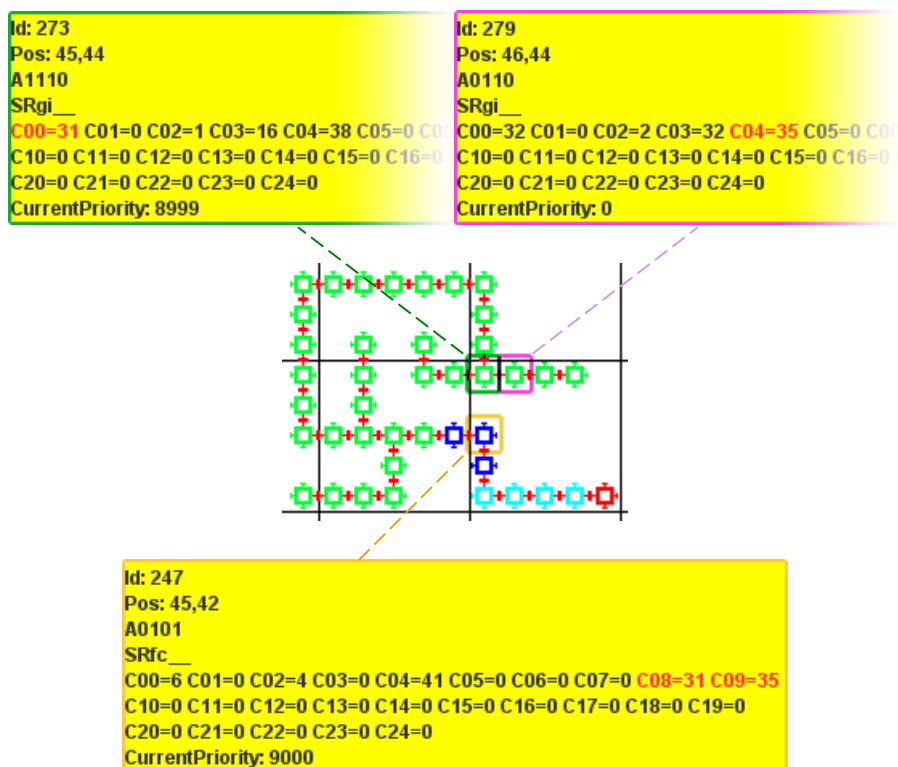
**Figura 7.12: Procés de detecció de forat “especial” i de rollback.** Es mostren diverses iteracions on s’observen la detecció de forats “especials” i el procés de rollback que aquests provoquen. El mòdul vermell identifica el màster i els mòduls taronja els líders de forat. Les zones marcades de taronja indiquen on es centra l’acció de cada iteració.



**7.7.4 Cul de sac:** un cul de sac es produeix quan dos mòduls provoquen un coll d'ampolla, en aquests casos un d'aquests mòduls ha d'evitar que mòduls externs al cul de sac hi entrin.

El mòdul responsable (líder del sac) és qui té un valor potencial més petit dels dos que formen el coll d'ampolla. Aquest mòdul és qui ha d'agafar les dades necessàries per controlar els mòduls que entren o salten.

A la Figura 7.13, el mòdul que realitza aquesta funció és el marcat amb taronja. El valor potencial d'aquest mòdul (*vpot* 6) és més petit que el del mòdul oposat (*vpot* 31). El líder del sac normalment agafa el valor potencial i el valor màxim del veí del sac, però en aquest exemple el cul de sac és de tipus "especial" (analitzat a l'Apartat 7.4.3). En aquest cas el valor potencial l'agafa del veí del sac (*vpot* 31), però el valor màxim l'agafa del mòdul veí de la branca externa, el mòdul marcat amb lila (*max* 35).



**Figura 7.13: Cul de sac "especial".** Es mostra una configuració amb un cul de sac "especial" i l'obtenció de les dades del líder del sac. El mòdul marcat de taronja és el líder del sac, el marcat de verd és el veí del sac i el mòdul marcat de lila és el veí de la bifurcació externa. El comptador C00 conté el valor potencial, el C04 el valor màxim i els C08 i C09 contenen la informació del sac del Nord.



## 8 Simulador

### 8.1 Descripció

La funció del simulador és la d'aplicar les regles descrites a l'arxiu *rules.txt* als mòduls que es troben a l'arxiu *agents.txt*.

Cada regla es divideix en 4 parts: el nom, la prioritat, la precondició i la postcondició.

El **nom de la regla** és un *string* no buit que ha de ser únic. Tota regla ha de tenir un nom diferent. L'objectiu del nom és facilitar la comprensió del codi a l'usuari.

La **prioritat** és un nombre enter, entre 1 i 32767, que determina en quin ordre s'executa la regla. Les regles que tenen una prioritat més alta s'executen abans. Si varies regles tenen la mateixa prioritat, si apliquen, s'executen alhora.

La **precondició** és un conjunt de condicions que s'han de complir per poder aplicar la regla. Entre condició i condició va una "AND" implícita, per tant, si una de les condicions no es compleix, la regla no s'aplica.

La **postcondició** és un conjunt d'accions que realitza el mòdul que aplica la regla.

L'estructura d'una regla és, doncs, la següent:

```
[nom]  
[prioritat]  
[precondició 1] [precondició 2] ... [precondició n]  
[postcondició 1] [postcondició 2] ... [postcondició m]
```

El simulador actua sobre tots els mòduls de forma sincronitzada. En cada iteració tots els mòduls verifiquen cadascuna de les regles i cada mòdul executa aquelles que s'apliquen a la seva situació (depenent de la prioritat i precondició) independentment de la resta de mòduls.

La situació inicial dels mòduls pot incloure certa informació: la seva posició inicial, el seu estat, les seves connexions amb els mòduls veïns i el valor de cadascun dels 25 comptadors.

La **posició inicial** són les coordenades absolutes en les quals es troba el mòdul. Aquesta dada és obligatòria.

L'**estat inicial** són els cinc caràcters que identifiquen en quin estat es troba el mòdul. Si no s'inicialitza, per defecte és indefinit S\*\*\*\*\*.

Les **connexions** determinen a quins veïns està connectat el mòdul. Si no s'indica res, un mòdul per defecte comença desconnectat de tots els veïns A0000.

Els **comptadors** són valors enters emmagatzemats a cada mòdul. Aquests valors poden anar de -32767 a 32767. Cada mòdul té 25 comptadors disponibles. Si no s'hi assigna cap valor, comencen inicialitzats a 0.

Per inicialitzar aquests valors, s'han de posar a continuació de la seva posició inicial separats per un espai. La forma d'inicialitzar cada valor és la següent:

[x,y] Posició global inicial

[S\_\_\_\_\_] Estat: inicialitza l'estat del mòdul. El valor pot ser {char, \*} un caràcter concret o un caràcter comodí (\*).

[A $\updownarrow\leftrightarrow$ \_\_\_\_\_] Connexions: inicialitza les connexions del mòdul amb els seus veïns (N, W, E, S) en aquest ordre. El valor referent a cada coordenada pot ser {0, 1}: 0 vol dir que està desconnectat i 1 que està connectat.

[C\_\_ \_\_\_\_\_] Comptadors: inicialitza el valor d'un dels comptadors del mòdul. El primer valor pot ser {0..24} i fa referència a quin dels comptadors es guarda la dada. El segon valor és la dada, un nombre enter.

Al fitxer agents.txt també s'indica la mida que té l'univers en el qual es mouen els mòduls. Aquest univers es delimita per coordenades absolutes indicant els límits de cada coordenada de la forma següent:

[U<sub>minX,maxX,minY,maxY</sub>] Límits de l'univers: indica el límits en els quals es poden moure els mòduls. Els valors són els nombres enters de cada coordenada mínima i màxima del pla x i y.

Tots els mòduls tenen informació pròpia:

- Id: *identificador que utilitza el sistema per diferenciar els mòduls.*
- Posició actual: *coordenades x, y absolutes d'on es troba.*
- Connexions: *a quins veïns està enganxat.*
- Estat: *estat actual del mòdul.*
- Comptadors: *25 comptadors disponibles per emmagatzemar valors.*
- Prioritat: *la prioritat de l'última regla aplicada.*
- Missatges de text: *4 possibles missatges rebuts, un per cada coordenada (N, W, E, S).*
- Missatges numèrics: *32 possibles missatges numèrics rebuts, 8 canals per cada coordenada (N, W, E, S).*

Durant el procés, el simulador permet consultar en qualsevol moment les dades actuals de qualsevol dels mòduls fent clic amb el ratolí sobre el mòdul en qüestió (podeu veure un exemple a la Figura 8.1). Si una dada no apareix significa que no té cap valor.

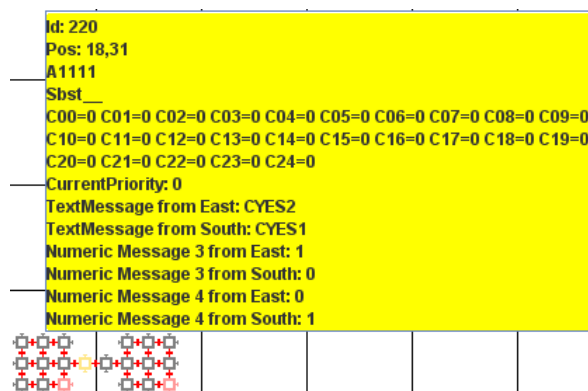


Figura 8.1: Informació d'un mòdul. Es mostra un exemple de la informació que pot proporcionar el simulador sobre l'estat actual d'un mòdul concret.

## 8.2 Funcionalitats

Per determinar què ha de fer un mòdul en cada situació, el simulador disposa de diferents funcionalitats que fan servir les regles, tan per a la comprovació de les condicions (precondicions), com per a la realització de les accions (postcondicions).

Amb el primer caràcter (en majúscula) és com s'identifica una funcionalitat. Aquest caràcter porta a continuació les dades que fa servir.

### · Precondicions:

- [**N**<sub>↕↔↕</sub>] Comprovació de veïns: *comprova en cada coordenada (N, W, E, S), en aquest ordre, si el mòdul té o no un veí. El valor referent a cada coordenada pot ser {0, 1, \*}: 0 indica que no hi ha veí, 1 que sí hi ha, \* que no importa si hi ha o no.*
- [**E**<sub>dx,dy</sub>] [**E**C<sub>\_\_,dy</sub>] Espais buits: *comprova que a les coordenades relatives dx, dy no hi hagi cap mòdul. La coordenada pot ser un nombre enter o el valor d'un dels comptadors del mòdul.*
- [**F**<sub>dx,dy</sub>] [**F**C<sub>\_\_,dy</sub>] Espais plens: *comprova que a les coordenades relatives dx, dy hi hagi un mòdul. La coordenada pot ser un nombre enter o el valor d'un dels comptadors del mòdul.*
- [**A**<sub>↕↔↕</sub>] Connexions: *comprova en cada coordenada (N, W, E, S), en aquest ordre, si el mòdul està connectat o no al seu veí. El valor referent a cada coordenada pot ser {0, 1, \*}: 0 indica que no està connectat, 1 que sí ho està, \* que no importa si està connectat o no.*
- [**S**<sub>\_\_\_\_\_</sub>] Estat: *comprova l'estat del mòdul caràcter per caràcter. El valor pot ser {char, \*} si és un caràcter el compara amb el de l'estat, si és \* vol dir que no importa el caràcter que hi ha en aquella posició.*
- [**T**<sub>dx,dy,\_\_\_\_\_</sub>] Estat d'un altre mòdul: *comprova l'estat del mòdul que es troba a les coordenades relatives dx, dy. El valor pot ser {char, \*} si és un caràcter el compara amb el de l'estat, si és \* vol dir que no importa el caràcter que hi ha en aquella posició.*
- [**M**<sub>±</sub> \_\_\_\_\_] Comprovació de missatges: *comprova el missatge de text que arriba al mòdul per la coordenada indicada. El primer valor pot ser {N, W, E, S, \*} una coordenades concreta o qualsevol d'elles (\*). El segon valor fa referència als cinc caràcters del missatge i poden ser {char, \*} si és un caràcter el compara amb el del missatge rebut, si és \* vol dir que no importa el caràcter que hi ha en aquella posició.*
- [**<**(-)\_\_\_\_\_ (-)\_\_\_\_\_] Comparació numèrica (més petit que): *comprova si el primer valor és més petit que el segon. El signe menys és opcional. Els valors poden ser {int, C<sub>\_\_\_</sub>, #<sub>\_\_\_</sub>} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.*
- [**>**(-)\_\_\_\_\_ (-)\_\_\_\_\_] Comparació numèrica (més gran que): *comprova si el primer valor és més gran que el segon. El signe menys és opcional. Els valors poden ser {int, C<sub>\_\_\_</sub>, #<sub>\_\_\_</sub>} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.*
- [**=**(-)\_\_\_\_\_ (-)\_\_\_\_\_] Comparació numèrica (igual que): *comprova si el primer valor és igual al segon. El signe menys és opcional. Els valors poden ser {int, C<sub>\_\_\_</sub>, #<sub>\_\_\_</sub>} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.*
- [**V**<sub>dx,dy,C<sub>\_\_\_</sub></sub> (-)\_\_\_\_\_] Comparació numèrica (comptador extern més petit que): *comprova si el primer valor és més petit que el segon. El signe menys és opcional. El primer valor fa referència a un comptador del mòdul que es troba a les coordenades relatives dx, dy. El segon valor pot ser {int, C<sub>\_\_\_</sub>, #<sub>\_\_\_</sub>} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.*

- [**W**<sub>dx,dy,C\_\_\_ (-)\_\_\_\_\_</sub>] Comparació numèrica (comptador extern més petit o igual que): comprova si el primer valor és més petit o igual que el segon. El signe menys és opcional. El primer valor fa referència a un comptador del mòdul que es troba a les coordenades relatives dx, dy. El segon valor pot ser {int, C\_\_\_, #\_\_\_} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.
- [**P**<sub>↑↔→↓</sub>] Prioritats: compara en cada coordenada (N, W, E, S), en aquest ordre, la prioritat del mòdul que indica la coordenada amb la del propi mòdul. El valor referent a cada coordenada pot ser {<, =, \*}: < indica que la prioritat del veí és més petita que la del mòdul, = que és més petita o igual, \* que no importa.
- [**L**<sub>dx,dy</sub>] [**LC**\_\_\_,dy] Prioritat més petita: comprova que la prioritat del mòdul que es troba a les coordenades relatives dx, dy sigui més petita. La coordenada pot ser un nombre enter o el valor d'un dels comptadors del mòdul.
- [**Q**<sub>dx,dy</sub>] [**QC**\_\_\_,dy] Prioritat més petita o igual: comprova que la prioritat del mòdul que es troba a les coordenades relatives dx, dy sigui més petita o igual. La coordenada pot ser un nombre enter o el valor d'un dels comptadors del mòdul.
- [**!**] Negació: inverteix el resultat de la precondició que segueix a la negació. Si la precondició dóna cert com a resultat el converteix a fals i viceversa.
- [**( )**] Parèntesis: agrupa diverses precondicions per poder tractar el resultat de totes elles com un únic resultat.

• Postcondicions:

- [**P**<sub>dx,dy</sub>] Canvi de posició (moviment): modifica la posició del mòdul a les coordenades relatives indicades. Tot i que les coordenades relatives poden ser qualsevol nombre enter, en el nostre sistema els valors utilitzats són {-1, 0, 1}.
- [**A**<sub>↑↔→↓</sub>] Unions: modifica les connexions del mòdul de cada coordenada (N, W, E, S), en aquest ordre. Les connexions i desconexions es fan en dues fases, una connexió/desconnexió prèvia al moviment, i una altra posterior. El valor referent a cada coordenada pot ser {0, 1, \*, +}: 0 es desconnecta i es manté desconnectat, 1 primer es desconnecta i després es connecta, \* es mantenen les connexions com estaven, + es manté connectat contínuament arrossegant amb ell els mòduls que tingui connectats.
- [**S**\_\_\_\_\_] Estat: modifica l'estat del mòdul caràcter per caràcter. El valor pot ser {char, \*} si és un caràcter es substitueix el que tenia, si és \* es manté el caràcter que tenia prèviament.
- [**M**<sub>↑</sub> \_\_\_\_\_] Enviament de missatges: envia un missatge de text per la coordenada indicada. El primer valor pot ser {N, W, E, S, \*} una coordenada concreta o totes elles (\*). El segon valor fa referència als cinc caràcters del missatge i poden ser {char, \*} un caràcter concret o un caràcter comodí (\*). Un missatge pot portar 4 caràcters comodí com a màxim.
- [**C**\_\_\_ - \_\_\_ - \_\_\_] Càlculs a comptadors: modifica el valor d'un dels comptadors del mòdul amb el resultat de l'operació entre dos operands. El primer valor indica el tipus d'operació, pot ser {+, -, \*, /, M, A, I}: els quatre primers realitzen les operacions aritmètiques bàsiques, M realitza el mòdul (residu de la divisió entera), A el valor màxim i I el valor mínim. Els operands poden ser {int, C\_\_\_, #\_\_\_} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.
- [**C**\_\_\_ - dx,dy,C\_\_\_ - \_\_\_] Càlculs a comptadors amb dades externes: com en el cas anterior, modifica el valor d'un dels comptadors del mòdul amb el resultat de l'operació entre dos operands. En aquest cas, qualsevol dels dos operands també pot ser el valor d'un dels comptadors del mòdul que es troba a les coordenades relatives dx, dy.

- [#\_\_\_ - \_\_\_\_ - \_\_\_\_] Càlculs a missatges numèrics: envia per un dels canals numèrics el resultat de l'operació entre dos operands. El primer valor indica el canal pel qual enviar el missatge (direcció i número de canal), aquest pot ser {N, W, E, S, \*} una coordenades concreta o totes elles (\*) i {01..08} el número de canal utilitzat. El segon indica el tipus d'operació, pot ser {+, -, \*, /, M, A, I}: els quatre primers realitzen les operacions aritmètiques bàsiques, M realitza el mòdul (residu de la divisió entera), A el valor màxim i I el valor mínim. Els operands poden ser {int, C\_\_\_, #\_\_\_} un nombre enter, el valor d'un dels comptadors del mòdul o el valor que li arriba per un dels canals numèrics.
- [#\_\_\_ - dx,dy,C\_\_\_ - \_\_\_\_] Càlculs a missatges numèrics amb dades externes: com en el cas anterior, envia per un dels canals el resultat de l'operació entre dos operands. En aquest cas, qualsevol dels dos operands també pot ser el valor d'un dels comptadors del mòdul que es troba a les coordenades relatives dx, dy.
- [X±] Swap (intercanvi): intercanvia la posició del mòdul amb la d'un mòdul veí. El valor pot ser {N, W, E, S} la coordenada on es troba el mòdul veí.

## 8.3 Modificacions

A mida que s'ha anat avançant en la implementació de les regles, es feien necessàries funcionalitats de les quals no disposàvem. Per aquesta raó, hem hagut de modificar part del codi del simulador.

S'han dut a terme 3 modificacions diferents: una per poder comparar comptadors propis amb comptadors d'altres mòduls, una altra per poder utilitzar els valors de comptadors d'altres mòduls en els càlculs a comptadors, i l'última per tenir més canals de comunicació disponibles.

A continuació es detalla el procés per a cadascuna de les modificacions realitzades. Aquestes modificacions s'han realitzat amb el programa *Eclipse* [16].

### 8.3.1 Noves funcionalitats: comparació numèrica a distància

Disposem de funcionalitats que permeten comparar valors numèrics que poden ser enters, valors extrets de comptadors del mòdul o valors de missatges numèrics rebuts. Aquestes funcionalitats són les que s'identifiquen per <, > o =. Però necessitàvem poder comparar dades d'altres mòduls.

En implementar les regles de la fase de reconfiguració a la forma canònica, tenim precondicions que necessiten obtenir informació de diferents mòduls. Aquesta informació es troba en els comptadors de cada mòdul. Per decidir un moviment, el mòdul necessita comparar el seu valor potencial i la seva branca amb la d'altres mòduls.

Les noves funcionalitats permeten comparar dades de comptadors d'altres mòduls (primer paràmetre), amb enters, valors de comptadors propis i missatges numèrics (segon paràmetre), aquestes funcionalitats són dues:

- [**V**<sub>dx,dy</sub>,C\_\_\_ (-)\_\_\_\_] Comparació numèrica (comptador extern més petit que).
- [**W**<sub>dx,dy</sub>,C\_\_\_ (-)\_\_\_\_] Comparació numèrica (comptador extern més petit o igual que).

Ambdues porten a terme el mateix procés per aconseguir el valor del comptador d'un altre mòdul. L'única diferència és, que una implementa la comparació "més petit que" i l'altra la "més petit o igual que"

Per implementar les noves funcionalitats ens hem basat en les funcionalitats esmentades anteriorment (<, >, =) ja que segueixen una estructura similar. Per aconseguir aquestes noves funcionalitats, s'han modificat els arxius següents:

- GlobalConstants.java: s'han afegit les constants *COUNTER\_SMALLER* = 'V' i *COUNTER\_SMALLER\_OR\_EQUAL* = 'W'.
- RuleManager.java: s'han afegit els casos de *COUNTER\_SMALLER* i *COUNTER\_SMALLER\_OR\_EQUAL*.
- RuleManagerAssistant.java: s'han afegit dues noves funcions *readAndStoreCounterSmallerField* i *readAndStoreCounterSmallerOrEqualField*. Aquestes funcions s'encarreguen tant d'extreure les dades necessàries per a la comparació i com de fer la comprovació d'errors de sintaxi.
- CounterFieldComparison.java: s'ha creat una nova classe que guarda el tipus de comparació a fer, els dos paràmetres a comparar i du a terme la comparació.
- RuleCheckerAssistant.java: s'han modificat les funcions *checkComparison* (s'ha afegit la opció de comparar enters) i *fulfillComparison* (s'han afegit els casos de *COUNTER\_SMALLER* i *COUNTER\_SMALLER\_OR\_EQUAL*).

### 8.3.2 Ampliació de funcionalitat: càlculs amb dades externes

En aquest cas, també disposem de funcionalitats que permeten operar amb valors numèrics (enters, valors de comptadors o de missatges numèrics) i assignar el resultat a un comptador o enviar-lo per un canal de comunicació. Aquestes funcionalitats són les que s'identifiquen per **C** o **#**. Però, de nou, necessitàvem poder operar amb dades d'altres mòduls.



En implementar les regles de la fase de reconfiguració a la forma canònica, tenim postcondicions que necessiten fer servir informació d'altres mòduls. En fer un moviment, el mòdul que el realitza necessita actualitzar el seu valor potencial. Aquest valor es determina en funció del valor potencial del mòdul al qual s'enganxa, per aquesta raó, necessita poder accedir al comptador que conté aquest valor.

Amb l'ampliació de les funcionalitats **C** i **#**, qualsevol dels operands pot ser: un enter, un valor d'un comptador, un valor de missatge, o un comptador d'un altre mòdul. Les funcionalitats afectades són les següents:

[**C**\_\_\_ \_ dx,dy,C\_\_\_ \_\_\_\_] Càlculs a comptadors amb dades externes.

[**#**\_\_\_ \_ dx,dy,C\_\_\_ \_\_\_\_] Càlculs a missatges numèrics amb dades externes.

L'obtenció d'aquestes dades externes té certes restriccions. Una són les coordenades dx i dy que no poden ser de més de 3 dígits. Per tant, el límit està en -999 i 999. L'altra és, que ens hem d'assegurar que aquell mòdul al qual volem accedir existeix per evitar agafar dades errònies. També cal vigilar que, si les dades les vol obtenir un mòdul en moviment, les coordenades relatives han de ser respecte a la seva posició final.

Les dues funcionalitats obtenen els valors dels operands de la mateixa forma, de tal manera que només es necessari fer una modificació. L'única diferència és on envien el resultat. Per aconseguir l'ampliació d'aquestes funcionalitats, s'han modificat els arxius següents:

- RuleManagerAssistant.java: s'ha modificat la funció *getOperand* afegint l'opció que l'operand pugui ser un comptador d'un altre mòdul. També realitza la comprovació d'errors de sintaxi.
- RulePerformerAssistant.java: s'ha modificat la funció *checkConsistency* afegint la verificació de la correctesa de la regla en cas de fer servir un comptador d'un altre mòdul. S'ha afegit la funció *checkNeighbourCounter* que és la que realitza aquesta verificació. S'ha modificat també la funció *getValue* afegint l'opció d'utilitzar un comptador d'un altre mòdul, i modificant tots els passos necessaris per aconseguir i retornar el valor d'aquest comptador.

### 8.3.3 Ampliació de canals: nous canals de comunicació

En aquest cas es tracta d'ampliar el nombre de canals pels quals els mòduls poden enviar missatges numèrics. Fins ara disposàvem de dos canals de comunicació per cada coordenada, aquest nombre es fa insuficient un cop volem l'execució de varies fases de forma paral·lela.

Un mòdul pot rebre informació de fases diferents i cadascuna es comunica per un canal concret, d'aquesta manera el mòdul podria gestionar diverses fases alhora. Tot i que aquesta situació no es pot donar, mantenim la divisió de canals per tenir un codi més net i facilitar la seva lectura.

De tota manera, un dels nous canals s'utilitza per transmetre el missatge de marca en la fase de construcció de la funció potencial.

Per aconseguir l'ampliació d'aquests canals, s'han modificat els arxius següents:

- *Agent2D.java: s'han afegit les variables CHANNEL\_3 a CHANNEL\_8, i NumericMessages3 a NumericMessage8. S'han modificat les funcions setNumericMessage, getNumericMessageValue i getNumericMessages afegint al codi els nous canals i nous missatges. També s'han modificat les funcions toString i copyNumericMessages per afegir els nous canals i la resetNumericMessages per netejar els nous missatges.*
- *Commander.java: s'ha modificat la funció getNumericMessages per afegir els nous canals.*
- *RuleManagerAssistant.java: s'han afegit les opcions dels nous canals a les funcions readAndStoreResult, getOperand, getParameter, isNumericAsteriskStrings i addCalculations.*
- *PostMessage.java: s'ha afegit a setChannel la opció dels nous canals.*
- *AgentPopupMenu.java: s'han afegit les variables de tots els missatges dels canals possibles (mnNPos3 ... mnNPos8; mnWPos3 ... mnWPos8; mnEPos3 ... mnEPos8; mnSPos3 ... mnSPos8) i s'ha modificat el lastPos a l'últim missatge, el mnSPos8. També s'ha modificat la funció storeValues amb els nous canals i els nous missatges que es poden generar.*

Les funcionalitats afectades per aquesta modificació són totes aquelles que poden fer servir canals de missatges numèrics: <, >, =, **V**, **W**, **C** i #.

## **8.4 A mode de llegenda**

Per tal de facilitar la comprensió d'allò que veiem en el simulador, en aquest apartat resumim la utilització que hem donat a cada estat, comptador i canal de comunicació en funció de la fase en què es troba un mòdul.

**Estats:**

[UNDEF]	Estat inicial.
[iniM0]	Candidat a màster ( <i>Master</i> ).
[iniCM]	Missatge de candidat a màster YES ( <i>Candidate Master</i> ).
[iniCS]	Missatge de candidat a màster NO ( <i>Candidate Slave</i> ).
[iniH0]	Candidat a líder de forat ( <i>Hole</i> ).
[iniCH]	Missatge de candidat a líder de forat YES ( <i>Candidate Hole</i> ).
[iniCL]	Missatge de candidat a líder de forat NO ( <i>Candidate Line</i> ).
[bst_M]	Màster en la fase build scan tree.
[bst_H]	Líder de forat en la fase build scan tree.
[bst_]	Resta de mòduls en la fase build scan tree.
[bpf_M]	Màster en la fase build potential function.
[bpfmM]	Màster en la fase build potential function (marcat).
[pfc_M]	Màster en la fase build potential function (completat).
[bpf_H]	Líder de forat en la fase build potential function.
[bpfmH]	Líder de forat en la fase build potential function (marcat).
[bpfwH]	Líder de forat en la fase build potential function (en espera).
[pfc_H]	Líder de forat en la fase build potential function (completat).
[bpf_]	Mòduls en espera del missatge de valor potencial.
[bpfm_]	Mòduls fent el recorregut de marcatge (anada).
[bpfw_]	Mòduls fent el recorregut de marcatge (en espera).
[pfc_]	Mòduls fent el recorregut de marcatge (tornada).
[pfcA_]	Mòduls fulla amb intenció d'activar-se.
[fw_A]	Mòduls actius en espera
[fwN_A]	Mòduls actius amb intenció de moure's al Nord
[fwS_A]	Mòduls actius amb intenció de moure's al Sud
[fwE_A]	Mòduls actius amb intenció de moure's a l'Est
[fwW_A]	Mòduls actius amb intenció de moure's a l'Oest
[fwNEA]	Mòduls actius amb intenció de moure's al Nord-Est
[fwSEA]	Mòduls actius amb intenció de moure's al Sud-Est
[fwNWA]	Mòduls actius amb intenció de moure's al Nord-Oest
[fwSWA]	Mòduls actius amb intenció de moure's al Sud-Oest
[fwdcA]	Mòduls actius decidint el camí a seguir (saltar o continuar)
[pfc_E]	Posicionat a la tira (End)
[Rgi_M]	Màster en la fase build final shape (informació disponible).
[Rgi_H]	Líder de forat en la fase build final shape (informació disponible).
[Rgi_]	Resta de mòduls en la fase build final shape (informació disponible).

\*Els estats "reverse" tenen els mateixos estats però amb 'R' al primer caràcter i els mateixos colors que la dels seus estats anàlegs.

## Comptadors:

	INI	BST	BPF	FW	BFS
C00	-	-	Valor potencial	Valor potencial	Valor potencial
C01	-	-	Signe	Signe	Responsable de forat
C02	-	-	Pare	Pare	Pare
C03	-	-	Mínim branca	Mínim branca	Mínim branca
C04	-	-	Màxim branca	Màxim branca	Màxim branca
C05	-	-	-	-	Posició de referència x
C06	-	-	-	-	Posició de referència y
C07	-	-	-	-	-
C08	-	-	-	-	Sac N (entrada)
C09	-	-	-	-	Sac N (salt)
C10	-	-	Direcció missatge	Mínim (opposite)	Sac O (entrada)
C11	-	-	Missatge canal #01	Màxim (opposite)	Sac O (salt)
C12	-	-	Missatge canal #02	Mínim (subbranca)	Sac E (entrada)
C13	-	-	-	Màxim (subbranca)	Sac E (salt)
C14	-	-	-	-	Sac S (entrada)
C15	-	-	-	-	Sac S (salt)
C16	-	-	-	-	Sac NW (entrada)
C17	-	-	-	-	Sac NW (salt)
C18	-	-	-	-	Sac NE (entrada)
C19	-	-	-	-	Sac NE (salt)
C20	-	-	-	Coordenada x	Sac SW (entrada)
C21	-	-	-	Coordenada y	Sac SW (salt)
C22	-	-	-	-	Sac SE (entrada)
C23	-	-	-	-	Sac SE (salt)
C24	-	-	-	Últim actiu	-

## Canals:

	INI	BST	BPF	FW	BFS
#01	-	-	Valor potencial	-	Valor potencial
#02	-	-	Mínim branca	-	Mínim branca
#03	Coord. x (candidate)	-	-	-	Coord. x (candidate)
#04	Coord. y (candidate)	-	-	-	Coord. y (candidate)
#05	-	-	Marcatge	-	Marcatge / Reset
#06	-	-	-	-	-
#07	-	-	-	-	-
#08	-	-	-	-	-

## 9 Gestió del projecte

### 9.1 Planificació i execució

Per realitzar la planificació inicial, el projecte es va dividir en 5 grans blocs, 2 d'elles opcionals:

1. **Programació:** aquest és el bloc principal del projecte. Inclou l'estudi del funcionament del simulador, l'estudi de la definició del projecte, la implementació de les regles i els jocs de proves.

Degut a la magnitud d'aquest bloc, decidim dividir-lo en diverses parts:

- 1.1. **Formació:** inclou l'estudi tant del funcionament del simulador com dels objectius i característiques del projecte.
  - 1.2. ***Finding a master & detecting the holes (simulació)*:** la implementació d'aquesta part es realitza externament a aquest projecte, per lo que s'ha de simular el seu efecte.
  - 1.3. ***Finding a master & detecting the holes (adaptació)*:** un cop incorporades les regles d'aquesta fase, es necessari adaptar-les a la resta del projecte. Inclou els jocs de proves.
  - 1.4. ***Building the scan tree*:** implementar les regles i realitzar els jocs de proves.
  - 1.5. ***Building the potential function*:** implementar les regles i realitzar els jocs de proves.
  - 1.6. ***Reconfiguring into a canonical shape (to master)*:** implementar les regles i realitzar els jocs de proves.
  - 1.7. ***Reconfiguring into a canonical shape (to strip)*:** implementar les regles i realitzar els jocs de proves.
  - 1.8. ***Building the final shape*:** implementar les regles i realitzar els jocs de proves. Aquesta part és opcional.
2. **Estudi de prototips:** aquest bloc es considera opcional. Es tracta de dur a terme un estudi de diferents prototips que s'acostin més al sistema que hem creat, de tal manera que es pugui arribar a utilitzar amb robots reals.
  3. **Creació de la pàgina web:** aquest bloc es considera opcional. Es tracta de crear un web amb tota la informació del projecte.
  4. **Elaboració de la memòria:** escriptura, modificació i correcció de la documentació del projecte.
  5. **Preparació de la presentació:** finalització i preparació de l'exposició del projecte.

També es tenen en compte les reunions periòdiques amb la directora del projecte. Els jocs de proves es realitzen acabar cadascuna de les fases del bloc 1.

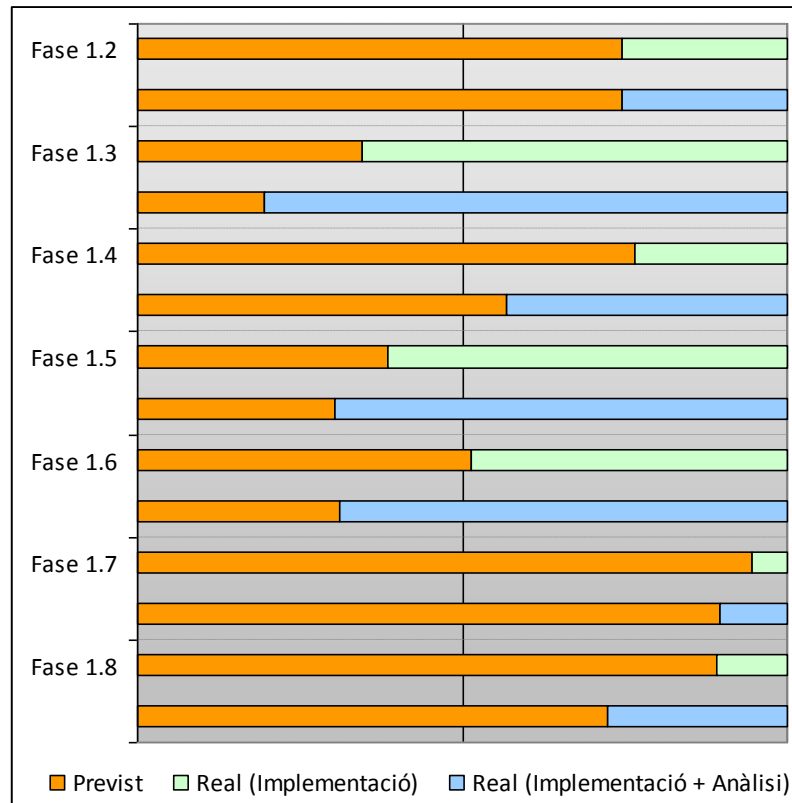
El càlcul de les hores es fa mitjançant una estimació molt general pel fet que, en tractar-se d'un treball vinculat a un projecte de recerca, en iniciar el projecte no es coneix quines complicacions es poden trobar. Totes les fases tenen un estudi teòric previ i això permet fer-se una idea de les dificultats potencials de cadascuna i assignar més o menys hores en funció d'aquesta complexitat o aprofitar per realitzar diverses tasques en paral·lel. En iniciar el projecte la idea general de la complexitat de les fases era la següent:

- Fase 1.2 (*Master & holes*): la simulació és trivial, a partir de treballs previs.
- Fase 1.3 (*Master & holes*): l'adaptació és fàcil de fer, però no obtindrem les regles fins al juliol (uns dos mesos des de l'inici del projecte).
- Fase 1.4 (*BST*): ràpid de fer.
- Fase 1.5 (*BPF*): fàcil d'implementar.
- Fase 1.6 (*FW Master*): la part més complicada i la que porta més feina.
- Fase 1.7 (*FW Strip*): un cop aconseguida la 1.6, aquesta ha de ser fàcil.
- Fase 1.8 (*BFS*): fàcil però laboriosa (opcional).

A la Figura 9.1 podem comprovar si la càrrega de treball que havíem previst en funció de la seva complexitat s'ajusta a la que finalment hem necessitat. Aquest gràfic està fet amb el percentatge d'hores de cada fase respecte la totalitat del projecte. El gràfic mostra si la previsió feta s'ajusta a la realitat o no.

Si la càrrega de la fase "prevista" i la "real" està equilibrada en el centre, vol dir que representen el mateix percentatge del total del projecte. Si el creuament es decanta a la dreta vol dir que el temps previst ha suposat un percentatge superior del projecte, en canvi, si és a l'esquerra vol dir que la càrrega de treball ha superat a la previsió feta.

La diferència en la previsió de la fase 1.8 no s'ha de tenir en consideració pel fet que només s'ha completat una part del total.



**Figura 9.1: Comparació de la previsió de les fases respecte la realitat.** Mostra en cada cas, si la previsió feta d'una fase es queda curta o no respecte al temps real. Es mostra la comparativa amb el temps emprat en la implementació de la fase, i amb el temps emprat per la implementació incloent el temps dedicat a la seva anàlisi.

Durant la realització de cada fase es segueix un procés metodològic que consisteix en el següent:

Primer es pensa com s'implementarà la fase, les possibles incidències que pot haver i les decisions que poden generar dubtes. Segon, juntament amb la directora del projecte, es discuteixen les opcions i els dubtes generats i es decideix com continuar. Tercer, s'implementa, i es realitzen proves per verificar el funcionament correcte. Quart, juntament amb la directora del projecte, es validen els resultats obtinguts o es realitzen les modificacions pertinents. Per últim, es documenten les coses més rellevants que s'han trobat durant la realització de la fase per tenir una referència a l'hora de redactar la memòria.

L'estimació inicial de la previsió de les fases dona lloc al calendari estimatiu de la Figura 9.2. Durant el transcurs del projecte s'ha anat readaptant aquest calendari a causa de les incidències i complicacions amb les quals ens hem anat trobant. Aquestes complicacions estan detallades en els apartats de Problemes de cada capítol.

A la Figura 9.3 podem observar com queda finalment el calendari amb el temps real dedicat a cadascuna de les fases un cop finalitzat el projecte.

Com s'observa en el calendari final, les fases inicials no s'han allunyat gaire de la previsió feta. D'aquestes fases teníem més informació i per tant hi ha hagut poques sorpreses. En canvi, les fases que realitzen el moviment (1.6) i l'adaptació de les regles de màster i forats (1.3) són les que han comportat majors problemes i les causants de les modificacions en el simulador. Aquestes fases, quan es va realitzar la planificació inicial, tan sols coneixíem allò que volíem que fessin però no com aconseguir-ho, això ha fet que la previsió no fos la més ajustada. A l'altre costat de la balança és troba la fase de generar la tira (1.7) que, finalment, ha resultat força més simple d'allò que s'esperava.

Aquestes possibles complicacions ja estaven previstes. Per prevenir aquests problemes es van incloure en el calendari parts opcionals, això donava la possibilitat de poder jugar amb elles i tenir temps de reserva per utilitzar en cas que fos necessari.

En el calendari final s'ha afegit una nova fase amb la que no es comptava en la previsió inicial. Aquesta fase és la que mostra el temps dedicat a la modificació del simulador. També s'ha marcat com a "Documentació" el temps invertit en l'estudi, anàlisi i documentació de les fases i les seves incidències. Parts d'aquesta documentació s'han aprofitat en la realització de la memòria.

En el diagrama de gantt de les figures 9.4 i 9.5 es pot observar com totes les fases són independents unes de les altres. L'inici de qualsevol de les fases es pot fer de forma simulada. Tot i que l'ordre establert facilita la comprensió global del projecte i l'obtenció de les dades dels mòduls de forma automàtica, es podria haver seguit un ordre totalment diferent. Canviar aquest ordre afegiria la complicació de simular tota la informació de cadascun dels mòduls i possibles problemes a l'hora de fer l'adaptació de cada fase.

A la Figura 9.6 es poden veure els percentatges del temps que ha comportat cadascuna de les fases respecte la totalitat del projecte.



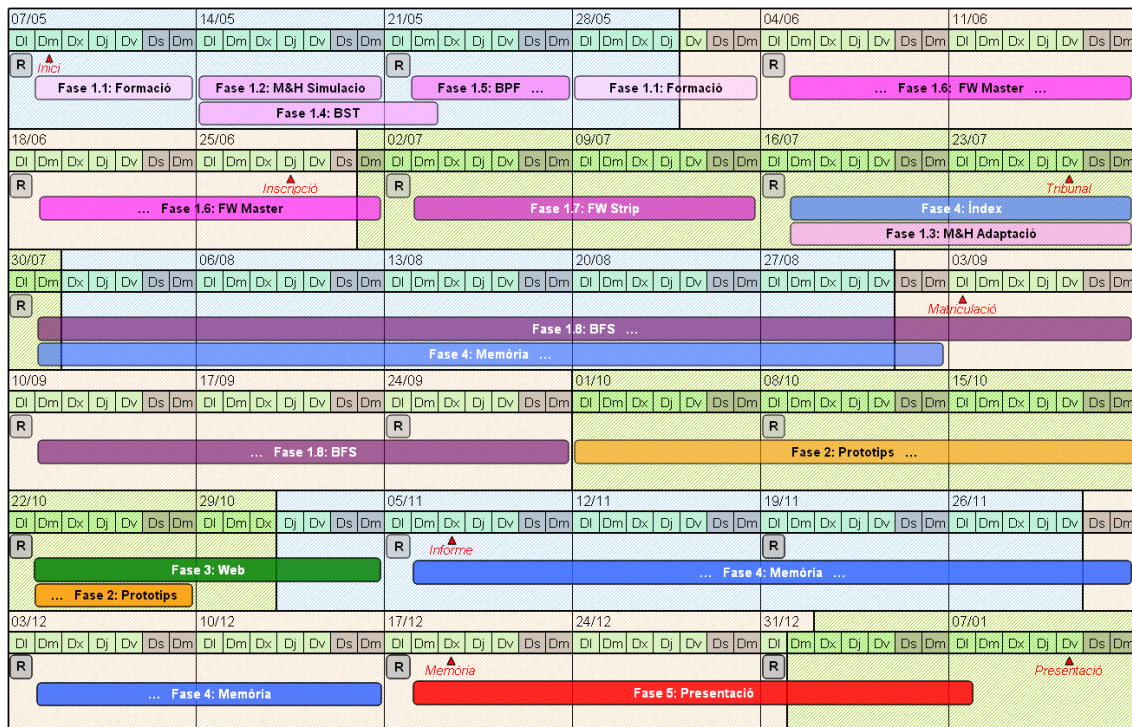


Figura 9.2: Calendari de la planificació inicial. Es mostra el calendari amb l'estimació de la durada de cada fase. Inclou les fites importants i les dates aproximades de les reunions.

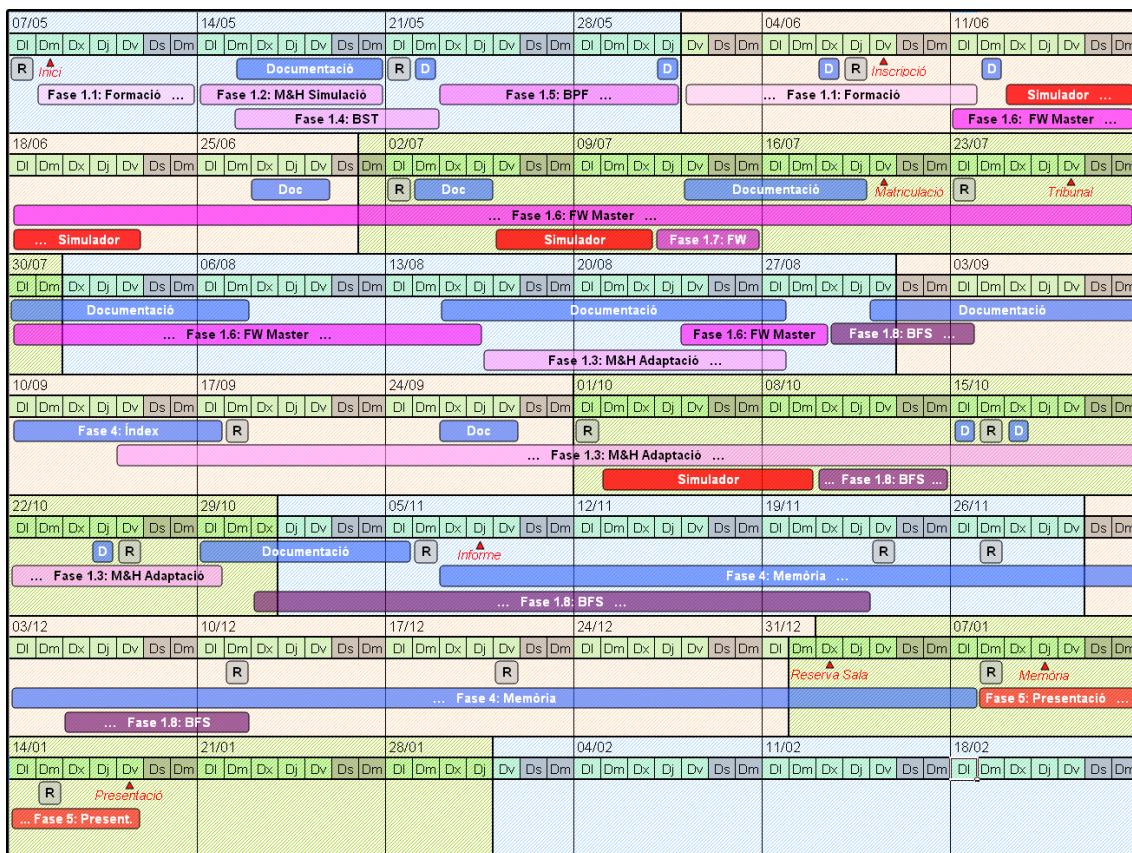
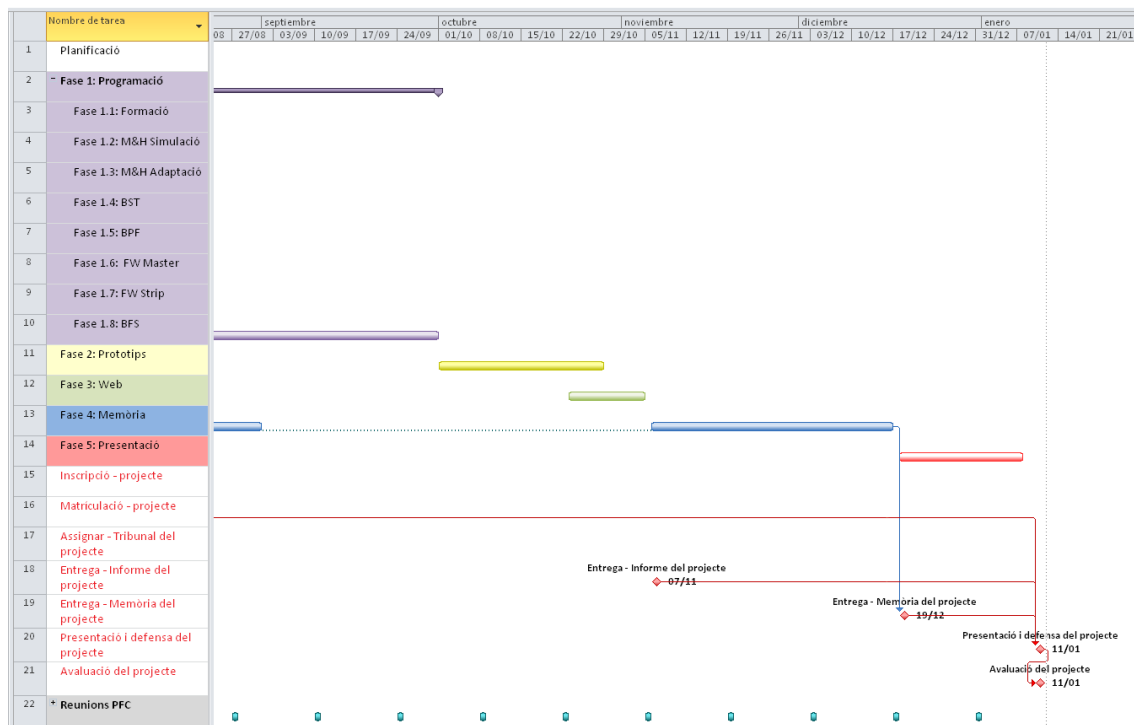
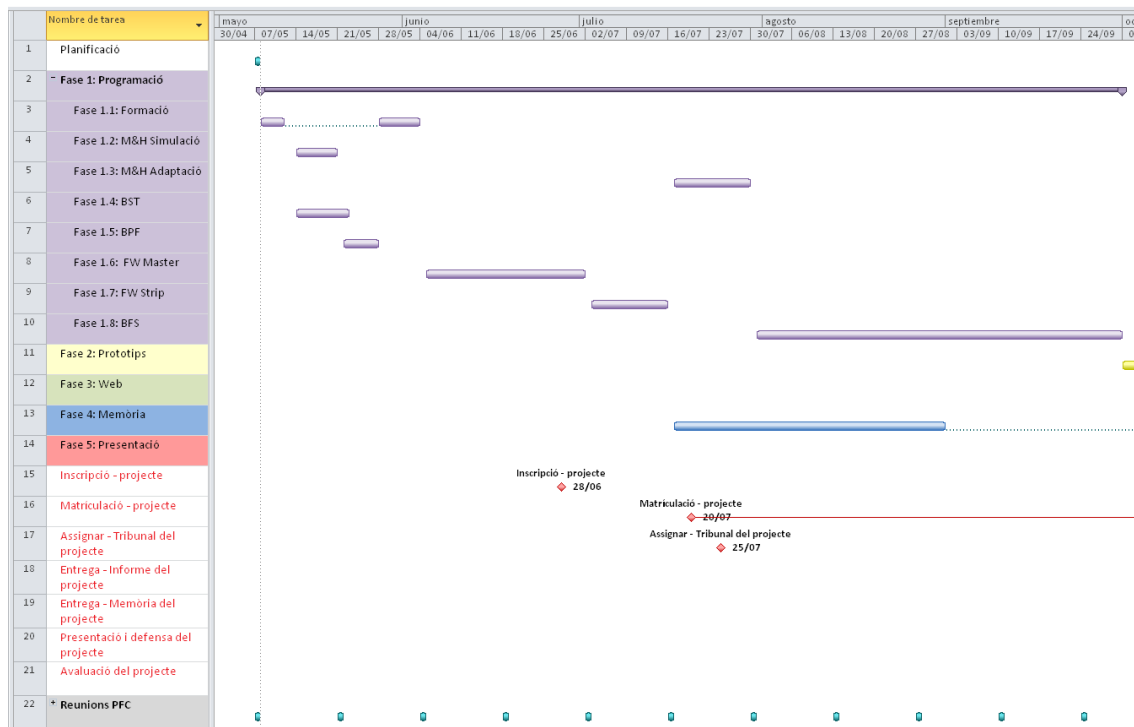
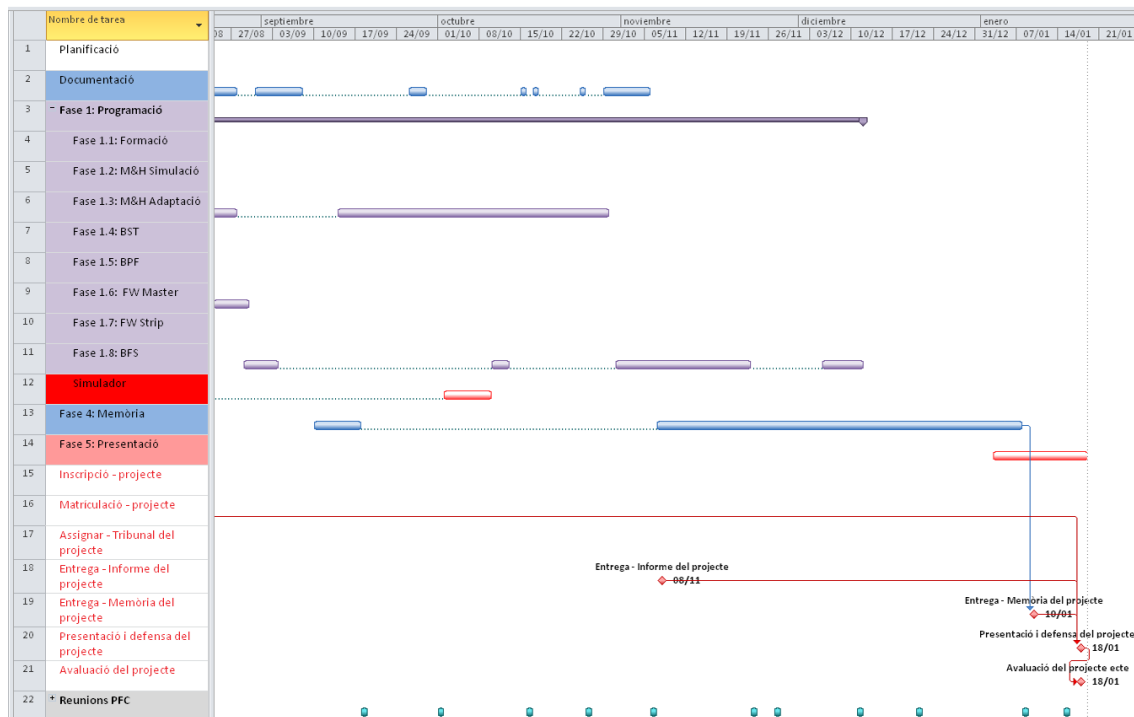
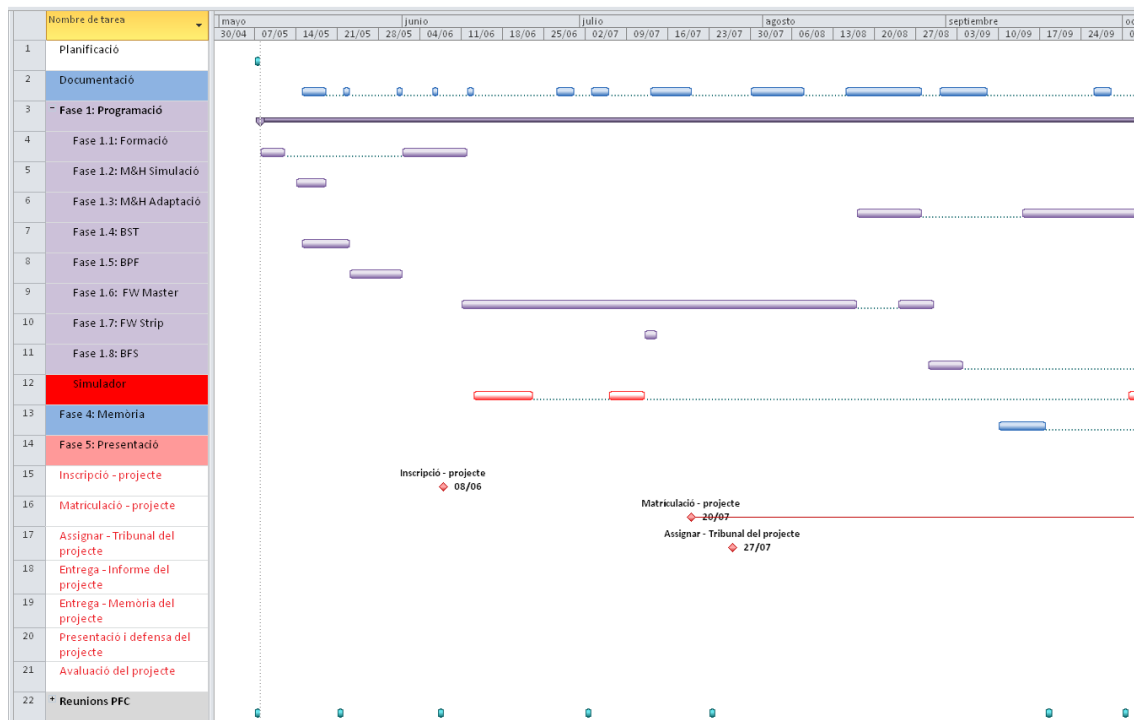


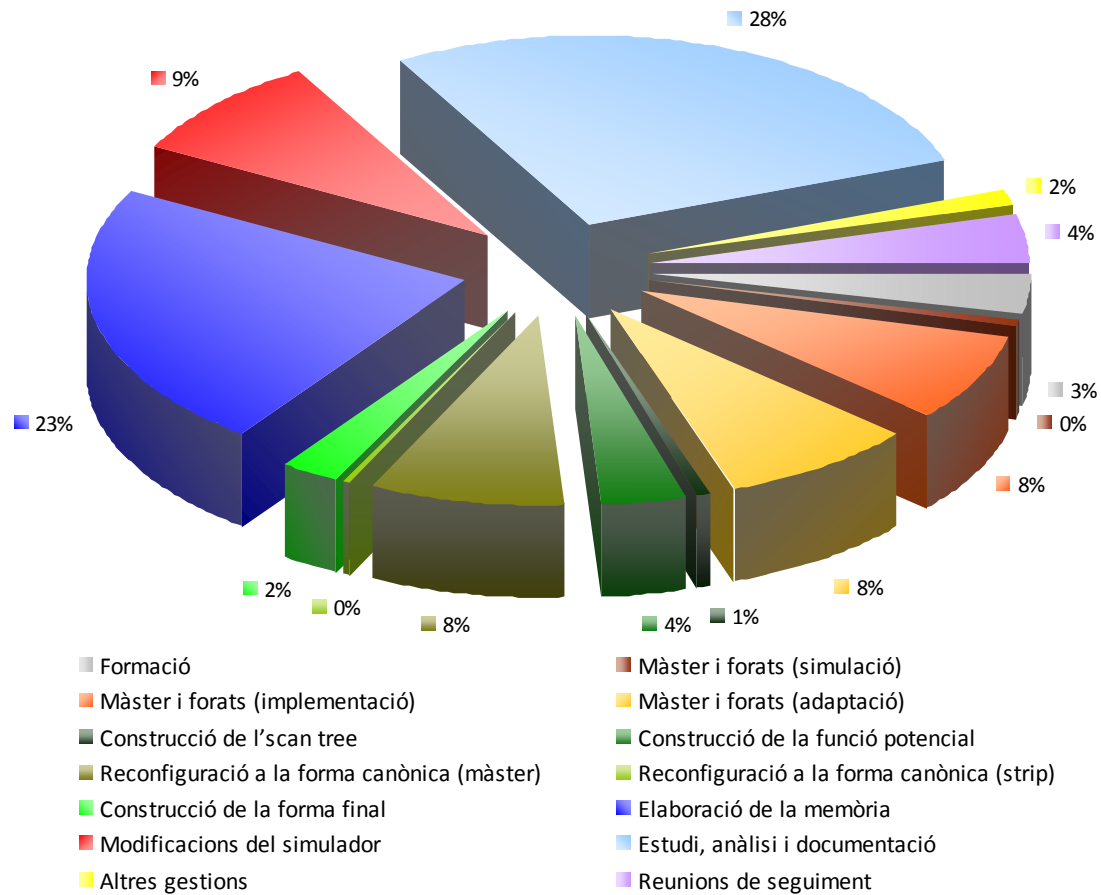
Figura 9.3: Calendari de l'execució. Es mostra el calendari amb la durada real de cada fase. Inclou les fites importants i les dates de les reunions realitzades.



**Figura 9.4 : Diagrama de gantt inicial.** Es mostra l'estimació de cascuna de les fases del projecte. Inclou les fites importants i les dates aproximades de les reunions.



**Figura 9.5: Diagrama de gantt final.** Es mostra la durada real de cadascuna de les fases del projecte. Inclou les fites importants i les dates de les reunions realitzades.



**Figura 9.6: Comparació de la càrrega de treball.** Es mostra el percentatge de temps que ha suposat cadascuna de les fases del projecte.

## 9.2 Pressupost

A continuació realitzem la valoració econòmica d'aquest projecte.

S'inclou el cost d'implementació de les primeres fases, que s'han realitzat externament. No s'inclou el cost del simulador, ni l'estudi previ pel fet que formen part d'altres projectes.

Tots els recursos utilitzats són humans. El material emprat ha estat digital per la qual cosa el seu cost no té rellevància en el pressupost.

En el projecte han intervingut 3 persones: la directora de projecte, l'analista i programador principal i un programador extern. La taula de la Figura 9.7 mostra el preu hora en funció del perfil professional.

Perfil	Preu/hora
Director de projecte	50 €
Analista	35 €
Programador	20 €

Figura 9.7: Preus segons el perfil professional.

A la taula de la Figura 9.8 es desglossen els recursos utilitzats, les hores emprades i els costos resultants de cadascuna de les fases.

Tasca	Hores	Recurs	Preu
Formació	27	Analista	945 €
Màster i forats (simulació)	3	Programador	60 €
Màster i forats (implementació)	60	Programador extern	1.200 €
Màster i forats (adaptació)	66	Programador	1.320 €
Construcció de l' <i>scan tree</i>	4	Programador	80 €
Construcció de la funció potencial	28	Programador	560 €
Reconfiguració a la forma canònica (màster)	66	Programador	1.320 €
Reconfiguració a la forma canònica ( <i>strip</i> )	2	Programador	40 €
Construcció de la forma final	19	Programador	380 €
Elaboració de la memòria	181	Analista	6.335 €
Modificacions del simulador	69	Programador	1.380 €
Estudi, anàlisi i documentació	219	Analista	7.665 €
Altres gestions	12	Analista	420 €
Reunions de seguiment	32	Director i Analista	2.720 €
<b>TOTAL</b>	<b>788</b>		<b>24.425 €</b>

Figura 9.8: Desglossament econòmic del projecte.



## **10 Conclusions**

### ***10.1 Resultats obtinguts***

L'objectiu principal del projecte era aconseguir, mitjançant regles, convertir una forma qualsevol d'un robot modular en una forma simple com una tira.

Aquest objectiu ha estat completat satisfactòriament i fins i tot s'han ampliat els objectius, tot aconseguint que els mòduls quedin preparats per una futura reconversió a una forma nova.

### ***10.2 Dificultats trobades***

La dificultat principal d'aquest projecte està en la quantitat de situacions diferents en les quals es pot trobar un mòdul i la necessitat de tenir totes aquestes possibilitats cobertes. Aquest fet ha incrementat, en molts casos, la complexitat de les regles per contemplar situacions que ni et plantejes que es puguin donar fins que no te les trobes. En això, el poder comptar amb un simulador ha estat essencial.

Per altra banda, implementar per mitjà de regles fa que per cada direcció del mòdul s'hagin d'implementar regles diferents tot i que conceptualment facin el mateix. Per exemple, no pots dir a un mòdul "ves endavant" li has de dir "ves al Nord" o "al Sud" o "a l'Est" o "a l'Oest". Això ha fet que la quantitat de regles a implementar es multipliqui. De la mateixa manera, algunes regles només modifiquen certes parts i mantenen la resta igual, o modifiquen les direccions d'on treuen les dades però l'estructura de les regles és manté.

Per ajudar en la implementació d'aquestes regles he fet un petit script que facilita la tasca de comprensió de les regles. Aquest script permet separar les parts d'una regla a la seva mínima expressió de manera que un llistat de precondicions o postcondicions es pot escriure per parts i l'script s'encarrega d'adaptar-ho a l'estil del simulador. Això facilita, a part de la comprensió visual en tenir cada condició per separat, poder fer servir, per exemple, (N-NE) en comptes de (1,2) evitant cometre errors més fàcilment; fer petits canvis d'una regla a una altra sense necessitat de reescriure tot; fer modificacions en una condició sense haver de buscar-la al llarg de tota la llista de condicions; etcètera.

Una altra de les complexitats ha estat aconseguir que un mòdul sigui capaç de conèixer l'estructura del robot i com fer-ho amb la mínima informació possible obtinguda del seu entorn local. Això ha estat crucial a l'hora d'evitar obstruccions en els colls d'ampolla, i també per a preparar l'estratègia de reconfiguració de la forma canònica a la forma final.

L'altre gran problema ha estat l'adaptació de les regles de les primeres fases a la resta del projecte pel fet que, en un principi, el procés començava de forma simultània. Trobar-nos que l'inici es produïa de forma asíncrona va fer que haguéssim d'adaptar la resta de les fases per contemplar la possibilitat que els mòduls haguessin d'esperar-se abans de continuar. Per tant, una cosa que semblava simple es va complicar substancialment.

Per últim, les modificacions en el simulador. Aquesta part es sortia de l'àmbit general del projecte i ens endinsava a un projecte diferent i en un llenguatge diferent. Per sort el codi implementat estava molt ben estructurat i documentat (*thanks Reinhard*) i ha estat més fàcil del que em pensava entendre el seu codi i realitzar les modificacions pertinents.

### 10.3 Futur del projecte

Per falta de temps no s'han pogut complir certes tasques que ens hauria agradat completar. Part d'aquestes tasques són les que s'havien proposat com a opcionals.

- **Reconfiguració a la forma final:** la idea que es tenia per ampliar el projecte era la d'aconseguir que el robot formés una nova forma a partir de la forma canònica. L'obtenció de la informació necessària i el raonament fet per aconseguir completar-ho es troben detallats en el Capítol 7.

- **Aplicació a prototips reals:** l'estudi de prototips era una altra de les parts opcionals. Tot el nostre projecte està basat en la simulació de robots. La finalitat és en un futur poder exportar lo que hem aconseguit a models de robots reals.

- **Creació de la pàgina web:** com a part opcional s'havia proposat organitzar tota la informació del projecte a una pàgina web. Aquesta idea es podria ampliar a aconseguir que, des de la web, qualsevol pogués accedir al simulador. Això podria permetre usar configuracions pròpies i veure el resultat en aplicar les regles o, fins i tot, donar l'opció de modificar o afegir noves regles i veure com es comporten els robots.

- **Extensió a 3D:** el sistema que hem realitzat està basat en les dues dimensions, per tant, totes les formes que podem representar són figures planes. En un futur es podria estendre les regles a models en tres dimensions contemplant una tercera coordenada.



## 10.4 Valoració personal

En termes generals estic molt satisfet d'allò que s'ha aconseguit, m'encanta veure els mòduls movent-se per la pantalla, saltant entre branques i esperant que el camí quedi lliure per avançar. Encara que sigui en un simulador... és impressionant!. És veritat que m'hauria agradat veure la part extra de transformar-se en una nova forma (que no hem pogut completar) funcionant i, posats a demanar, veure tot en funcionament amb models reals. El temps no ha donat per més, una llàstima. Espero que la feina feta serveixi per donar un passet més en el món dels robots modulars, crec que és una branca amb molt potencial.

Fins ara mai no havia realitzat un projecte de tal magnitud, no només pel temps i l'esforç de dedicació, sinó també per ser una part d'un projecte molt més gran. Molta feina que hi ha darrera abans que jo comencés i, segurament, molta feina que encara quedarà per fer. També és un tipus de projecte en desenvolupament, on no se sap amb certesa com fer les coses, s'ha d'anar adaptant a mida que s'avança i això dificulta la seva planificació. Tots els treballs que he anat fent al llarg de la carrera tenien un inici i un final definits, en canvi, aquest tipus de projecte et fa plantejar-te les coses d'una manera diferent.

Les habilitats adquirides durant la carrera m'han ajudat a dur a terme aquesta tasca, no tant a nivell de coneixements (que també), sinó més com a conceptes generals, "l'estil" de fer les coses: com tractar un projecte gran, com dividir la feina, com organitzar-te, com raonar els problemes, ...

També m'he adonat de la dificultat de documentar aquest tipus de treball. Jo sóc molt de puntualitzar coses concretes, fer esquemes o resums, però no sóc d'escriure grans parrafades. Això també era pel fet que fins ara els resums eren per mi i amb unes quantes dades era suficient per recordar-me del context. En canvi, aquest cop he provat d'explicar les coses esplaïant-me més, de manera que fos més fàcil que una persona que no coneix en profunditat el tema el pogués entendre. És difícil tenint en compte els tecnicismes associats a aquest tipus de projecte, espero haver-ho aconseguit.

Per últim, vull agrair a la Vera Sacristán i la seva implicació en el projecte. Facilita moltíssim la feina i redueix l'estrès saber que hi ha una persona implicada al darrera que t'ajuda i et guia quan et trobes una mica perdut i no saps per on continuar.

En resum, ha estat una bona experiència i he gaudit més del que em pensava.

Gràcies.



## Referències

- [1] Juan González Gómez, Robótica modular y Locomoción: Aplicación a Robots Ápodos, *Universidad Autónoma de Madrid, Thesis*, 2008.
- [2] T. Fukuda, T. Ueyama, Y. Kawauchi, F. Arai. Concept of cellular robotic system (CEBOT) and basic strategies for its realization. *Computers and Electronical Engineering*, 18(1):11-39, 1992.
- [3] A. Pamecha, C. Chiang, D. Stein, G. Chirikjian. Design and implementation of metamorphic robots. In *Proc. of the ASME Design Engineering Technical Conference and Computers in Engineering Conference*, 1996.
- [4] S. Murata, H. Kurokawa, E. Yoshida, K. Tomita, S. Kokaji. A 3-D self-reconfigurable structure. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 432-439, 1998.
- [5] D. Rus, M. Vona. Crystalline robots: Self-reconfiguration with compressible unit modules. *Autonomous Robots*, 10(1):107-124, 2001.
- [6] Z. Butler, R. Fitch, D. Rus. Distributed control for unit-compressible robots: Goal-recognition, locomotion and splitting. *IEEE/ASME Transactions on Mechatronics*, 7(4):418-430, 2002.
- [7] M. W. Jorgensen, E. H. Esben, H. Ostergaard, H. H. Lund. Modular ATRON: Modules for a self-reconfigurable robot. In *Proc. of IEEE/RSJ International Conference on Robots and Systems (IROS)*, pages 2068-2073, 2004.
- [8] K. Gilpin, K. Kotay, D. Rus, I. Vasilescu. Miche: Modular shape formation by self-disassembly. *The International Journal of Robotics Research*, 27(3-4):345-372, 2008.
- [9] Byoung Kwon An. EM-Cube: Cube-Shaped, Self-reconfigurable Robots Sliding on Structure Surfaces. In *IEEE International Conference on Robotics and Automotion (ICRA)*, pages 3149-3155, 2008.
- [10] Cheung, K. C., Demaine, E. D., Bachrach, J. R., and Griffith, S., Programmable Assembly With Universally Foldable Strings (Moteins), *IEEE Transactions on Robotics*, vol. 27, no. 4, pp. 718-729, 2011.
- [11] H. Kurokawa, K. Tomita, A. Kamimura, S. Kokaji, T. Hasuo, S. Murata. Distributed self-reconfiguration of M-TRAN III modular robotic system. *The International Journal of Robotics Research*, 27:373-386, 2008.

- [12] Oswin Aichholzer, Thomas Hackl, Vera Sacristán, Birgit Vogtenhuber, Reinhard Wallner, Self-organizing lattice-based modular robotic systems, *manuscript*.
- [13] Ferran Hurtado, Enrique Molina, Suneeta Ramaswami, Vera Sacristán, Generic distributed universal constant-force reconfiguration of 2D lattice-based modular robotic systems, *manuscript*.
- [14] Reinhard Wallner, A System of Autonomously Self-Reconfigurable Agents, *Diploma Thesis, Institute for Software Technology, Graz. University of Technology*, 2009.
- [15] A. Sproewitz, A. Billard, P. Dillenbourg, A. J. Ijspeert. Roombots-mechanical design of self-reconfiguring modular robots for adaptive furniture. In *Proc. of the IEEE International Conference on Robotics and Automation (ICRA)*, pages 4259-4264, 2009.
- [16] The Eclipse Foundation, Eclipse IDE for Java Developers, <http://www.eclipse.org>.

# ***Annex***



# Annexos

<b>A. Regles de la fase de construcció de l'scan tree .....</b>	<b>125</b>
<i>Build scan tree Hole .....</i>	<i>126</i>
<i>Build scan tree Not-East .....</i>	<i>127</i>
<i>Build scan tree Not-West .....</i>	<i>128</i>
<i>Build scan tree Others .....</i>	<i>129</i>
 <b>B. Regles de la fase de construcció de la funció potencial .....</b>	 <b>131</b>
<b>1. Grup "start" .....</b>	<b>131</b>
<i>Start potential function to N .....</i>	<i>132</i>
<i>Start potential function to N new branch .....</i>	<i>133</i>
<i>Start potential function to W .....</i>	<i>134</i>
<b>2. Grup "first go" .....</b>	<b>135</b>
<i>Spread potential function first from S to E same branch .....</i>	<i>136</i>
<i>Spread potential function first from S to E new branch .....</i>	<i>137</i>
<i>Spread potential function first from S to N same branch .....</i>	<i>138</i>
<i>Spread potential function first from S to N new branch .....</i>	<i>139</i>
<i>Spread potential function first from S to W same branch .....</i>	<i>140</i>
<i>Spread potential function first last child S to S .....</i>	<i>141</i>
<i>Spread potential function first from E to N same branch .....</i>	<i>142</i>
<i>Spread potential function first from E to N new branch .....</i>	<i>143</i>
<i>Spread potential function first from E to W same branch .....</i>	<i>144</i>
<i>Spread potential function first from E to W new branch .....</i>	<i>145</i>
<i>Spread potential function first from E to S same branch .....</i>	<i>146</i>
<i>Spread potential function first last child E to E .....</i>	<i>147</i>
<i>Spread potential function first from N to W same branch .....</i>	<i>148</i>
<i>Spread potential function first from N to W new branch .....</i>	<i>149</i>
<i>Spread potential function first from N to S same branch .....</i>	<i>150</i>
<i>Spread potential function first from N to S new branch .....</i>	<i>151</i>
<i>Spread potential function first from N to E same branch .....</i>	<i>152</i>
<i>Spread potential function first last child N to N .....</i>	<i>153</i>
<i>Spread potential function first from W to S same branch .....</i>	<i>154</i>
<i>Spread potential function first from W to S new branch .....</i>	<i>155</i>
<i>Spread potential function first from W to E same branch .....</i>	<i>156</i>
<i>Spread potential function first from W to E new branch .....</i>	<i>157</i>
<i>Spread potential function first from W to N same branch .....</i>	<i>158</i>
<i>Spread potential function first last child W to W .....</i>	<i>159</i>
<b>3. Grup "first wait" .....</b>	<b>160</b>
<i>Spread potential function first from S wait .....</i>	<i>161</i>
<i>Spread potential function first from E wait .....</i>	<i>162</i>
<i>Spread potential function first from N wait .....</i>	<i>163</i>
<i>Spread potential function first from W wait .....</i>	<i>164</i>

<b>4. Grup “wait-continue” .....</b>	<b>165</b>
<i>Spread potential function first from S to E same branch cont .....</i>	<i>166</i>
<i>Spread potential function first from S to E new branch cont .....</i>	<i>167</i>
<i>Spread potential function first from S to N same branch cont .....</i>	<i>168</i>
<i>Spread potential function first from S to N new branch cont .....</i>	<i>169</i>
<i>Spread potential function first from S to W same branch cont .....</i>	<i>170</i>
<i>Spread potential function first last child S to S cont .....</i>	<i>171</i>
<i>Spread potential function first from E to N same branch cont .....</i>	<i>172</i>
<i>Spread potential function first from E to N new branch cont .....</i>	<i>173</i>
<i>Spread potential function first from E to W same branch cont .....</i>	<i>174</i>
<i>Spread potential function first from E to W new branch cont .....</i>	<i>175</i>
<i>Spread potential function first from E to S same branch cont .....</i>	<i>176</i>
<i>Spread potential function first last child E to E cont .....</i>	<i>177</i>
<i>Spread potential function first from N to W same branch cont .....</i>	<i>178</i>
<i>Spread potential function first from N to W new branch cont .....</i>	<i>179</i>
<i>Spread potential function first from N to S same branch cont .....</i>	<i>180</i>
<i>Spread potential function first from N to S new branch cont .....</i>	<i>181</i>
<i>Spread potential function first from N to E same branch cont .....</i>	<i>182</i>
<i>Spread potential function first last child N to N cont .....</i>	<i>183</i>
<i>Spread potential function first from W to S same branch cont .....</i>	<i>184</i>
<i>Spread potential function first from W to S new branch cont .....</i>	<i>185</i>
<i>Spread potential function first from W to E same branch cont .....</i>	<i>186</i>
<i>Spread potential function first from W to E new branch cont .....</i>	<i>187</i>
<i>Spread potential function first from W to N same branch cont .....</i>	<i>188</i>
<i>Spread potential function first last child W to W cont .....</i>	<i>189</i>
<b>5. Grup “continue” .....</b>	<b>190</b>
<i>Spread potential function continue from S to E .....</i>	<i>191</i>
<i>Spread potential function continue from S to N .....</i>	<i>192</i>
<i>Spread potential function continue from E to N .....</i>	<i>193</i>
<i>Spread potential function continue from E to W .....</i>	<i>194</i>
<i>Spread potential function continue from N to W .....</i>	<i>195</i>
<i>Spread potential function continue from N to S .....</i>	<i>196</i>
<i>Spread potential function continue from W to S .....</i>	<i>197</i>
<i>Spread potential function continue from W to E .....</i>	<i>198</i>
<b>6. Grup “last” .....</b>	<b>199</b>
<i>Spread potential function last from S to E .....</i>	<i>200</i>
<i>Spread potential function last from S to N .....</i>	<i>201</i>
<i>Spread potential function last from S to W .....</i>	<i>202</i>
<i>Spread potential function last from E to N .....</i>	<i>203</i>
<i>Spread potential function last from E to W .....</i>	<i>204</i>
<i>Spread potential function last from E to S .....</i>	<i>205</i>
<i>Spread potential function last from N to W .....</i>	<i>206</i>
<i>Spread potential function last from N to S .....</i>	<i>207</i>
<i>Spread potential function last from N to E .....</i>	<i>208</i>
<i>Spread potential function last from W to S .....</i>	<i>209</i>
<i>Spread potential function last from W to E .....</i>	<i>210</i>
<i>Spread potential function last from W to N .....</i>	<i>211</i>
<b>7. Grup “last father” .....</b>	<b>212</b>
<i>Spread potential function last father from N .....</i>	<i>213</i>
<i>Spread potential function last father from W .....</i>	<i>214</i>



8. Grup extra “first go adapt” i “first wait adapt” .....	215
<i>Spread potential function first from S to E same branch (ini adapt)</i> .....	216

## C. Regles de la fase de reconfiguració a la forma canònica ..... 217

1. Grup “activate” .....	217
<i>Activate Leaf N 1</i> .....	218
<i>Activate Leaf W 1</i> .....	219
<i>Activate Leaf S 1</i> .....	220
<i>Activate Leaf E 1</i> .....	221
<i>Activate Leaf N 2</i> .....	222
<i>Activate Leaf W 2</i> .....	223
<i>Activate Leaf S 2</i> .....	224
<i>Activate Leaf E 2</i> .....	225
<i>Activate Leaf cant activate</i> .....	226
2. Grup “slide” .....	227
<i>Slide E 1</i> .....	228
<i>Slide E 2 cant move</i> .....	229
<i>Slide E 2 win active</i> .....	230
<i>Slide E 2 win E-NE</i> .....	231
<i>Slide E 2 win E-E</i> .....	232
<i>Slide E 2 win E-SE</i> .....	233
<i>Slide E 2 win</i> .....	234
<i>Slide N 1</i> .....	235
<i>Slide N 2 cant move</i> .....	236
<i>Slide N 2 win active</i> .....	237
<i>Slide N 2 win N-NW</i> .....	238
<i>Slide N 2 win N-N</i> .....	239
<i>Slide N 2 win N-NE</i> .....	240
<i>Slide N 2 win</i> .....	241
<i>Slide W 1</i> .....	242
<i>Slide W 2 cant move</i> .....	243
<i>Slide W 2 win active</i> .....	244
<i>Slide W 2 win W-NW</i> .....	245
<i>Slide W 2 win W-W</i> .....	246
<i>Slide W 2 win W-SW</i> .....	247
<i>Slide W 2 win</i> .....	248
<i>Slide S 1</i> .....	249
<i>Slide S 2 cant move</i> .....	250
<i>Slide S 2 win active</i> .....	251
<i>Slide S 2 win S-SW</i> .....	252
<i>Slide S 2 win S-S</i> .....	253
<i>Slide S 2 win S-SE</i> .....	254
<i>Slide S 2 win</i> .....	255

<b>3. Grup “convex transition” .....</b>	<b>256</b>
Convex Transition SE 1 .....	257
Convex Transition SE 2 cant move .....	258
Convex Transition SE 2 win NE .....	259
Convex Transition SE 2 win E .....	260
Convex Transition SE 2 win SE .....	261
Convex Transition SE 2 win S .....	262
Convex Transition SE 2 win SW .....	263
Convex Transition SE 2 win .....	264
Convex Transition NE 1 .....	265
Convex Transition NE 2 cant move .....	266
Convex Transition NE 2 win NW .....	267
Convex Transition NE 2 win N .....	268
Convex Transition NE 2 win NE .....	269
Convex Transition NE 2 win E .....	270
Convex Transition NE 2 win SE .....	271
Convex Transition NE 2 win .....	272
Convex Transition NW 1 .....	273
Convex Transition NW 2 cant move .....	274
Convex Transition NW 2 win SW .....	275
Convex Transition NW 2 win W .....	276
Convex Transition NW 2 win NW .....	277
Convex Transition NW 2 win N .....	278
Convex Transition NW 2 win NE .....	279
Convex Transition NW 2 win .....	280
Convex Transition SW 1 .....	281
Convex Transition SW 2 cant move .....	282
Convex Transition SW 2 win SE .....	283
Convex Transition SW 2 win S .....	284
Convex Transition SW 2 win SW .....	285
Convex Transition SW 2 win W .....	286
Convex Transition SW 2 win NW .....	287
Convex Transition SW 2 win .....	288
 <b>4. Grup “concave transition” .....</b>	 <b>289</b>
Concave Transition S to E diff branch .....	290
Concave Transition E to N diff branch .....	291
Concave Transition N to W diff branch .....	292
Concave Transition W to S diff branch .....	293
Concave Transition S to E same branch better Pot .....	294
Concave Transition E to N same branch better Pot .....	295
Concave Transition N to W same branch better Pot .....	296
Concave Transition W to S same branch better Pot .....	297
Concave Transition S to E same branch worse Pot .....	298
Concave Transition E to N same branch worse Pot .....	299
Concave Transition N to W same branch worse Pot .....	300
Concave Transition W to S same branch worse Pot .....	301

<b>5. Grup “opposite transition”</b>	<b>302</b>
<i>Opposite Transition S to N diff branch</i>	304
<i>Opposite Transition E to W diff branch</i>	305
<i>Opposite Transition N to S diff branch</i>	306
<i>Opposite Transition W to E diff branch</i>	307
<i>Opposite Transition S to N same branch better Pot</i>	308
<i>Opposite Transition E to W same branch better Pot</i>	309
<i>Opposite Transition N to S same branch better Pot</i>	310
<i>Opposite Transition W to E same branch better Pot</i>	311
<i>Opposite Transition S to N same branch worse Pot</i>	312
<i>Opposite Transition E to W same branch worse Pot</i>	313
<i>Opposite Transition N to S same branch worse Pot</i>	314
<i>Opposite Transition W to E same branch worse Pot</i>	315
<i>Opposite Transition S to N same branch worse Pot 1</i>	316
<i>Opposite Transition S to N same branch worse Pot 2 win active</i>	317
<i>Opposite Transition S to N same branch worse Pot 2 jump</i>	318
<i>Opposite Transition E to W same branch worse Pot 1</i>	319
<i>Opposite Transition E to W same branch worse Pot 2 win active</i>	320
<i>Opposite Transition E to W same branch worse Pot 2 jump</i>	321
<i>Opposite Transition N to S same branch worse Pot 1</i>	322
<i>Opposite Transition N to S same branch worse Pot 2 win active</i>	323
<i>Opposite Transition N to S same branch worse Pot 2 jump</i>	324
<i>Opposite Transition W to E same branch worse Pot 1</i>	325
<i>Opposite Transition W to E same branch worse Pot 2 win active</i>	326
<i>Opposite Transition W to E same branch worse Pot 2 jump</i>	327
<i>Opposite Transition * to * same branch worse Pot 2 cant jump</i>	328
<b>6. Grup “strip”</b>	<b>329</b>
<i>Complete Strip</i>	330
<i>Send Mark Leaf Strip</i>	331
<i>Receive Mark Leaf Strip</i>	332
<b>D. Regles de la fase de construcció de la forma final</b>	<b>333</b>
<b>1. Grup “reverse rules”</b>	<b>333</b>
<i>[R] Spread potential function first from S to W new branch</i>	334
<b>2. Grup “rollback”</b>	<b>335</b>
<i>[R] Start rollback (special hole detected)</i>	336
<i>[R] Spread rollback master</i>	337
<i>[R] Spread rollback master extra</i>	338
<i>[R] Spread rollback others</i>	339

3. Grup “get info” .....	340
[R] Get bottleneck info from N .....	342
[R] Get bottleneck info from W .....	343
[R] Get bottleneck info from E .....	344
[R] Get bottleneck info from S .....	345
[R] Get bottleneck info from N special .....	346
[R] Get bottleneck info from W special .....	347
[R] Get bottleneck info from E special .....	348
[R] Get bottleneck info from S special .....	349
[R] Get bottleneck info from NW .....	350
[R] Get bottleneck info from NE .....	351
[R] Get bottleneck info from SW .....	352
[R] Get bottleneck info from SE .....	353
[R] Get bottleneck info complete .....	354
[R] Get hole responsible Type 1 .....	355
[R] Get hole responsible Type 2 .....	356

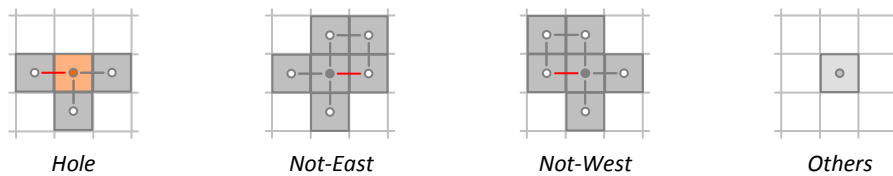
## Annex A

### *Regles de la fase de construcció de l'scan tree*

En aquest apartat es detallen les regles utilitzades en la fase de construcció de l'arbre [bst] (*Build Scan Tree*). L'objectiu, l'estratègia i una visió general d'aquest conjunt de regles es descriuen al Capítol 4.

En aquest apartat hem implementat les regles següents:

- Build scan tree



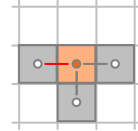
## Annex A - Regles de la fase de construcció de l'scan tree

---

<b>Regla:</b>	<i>Build scan tree Hole</i>
<b>Prioritat:</b>	<i>10000</i>
<b>Precondicions:</b>	<i>Sbst**</i> <i>S****H</i>
<b>Postcondicions:</b>	<i>A*0**</i>

- **Regla:** Build scan tree Hole

*Aquesta regla fa referència als mòduls identificats com a líders de forat que es troben en la fase bst.*



- **Prioritat:** 10000

*La prioritat és la mateixa que la de les fases [ini] i [bpf] per poder treballar en paral·lel.*

*La prioritat és la mateixa que la resta de regles de la fase. Això permet aplicar aquesta regla i la de canvi d'estat simultàniament.*

- **Precondicions:**

*Sbst\*\* : afecta els mòduls de la fase [bst]*

*S\*\*\*\*H : afecta els mòduls identificats com a líders de forat.*

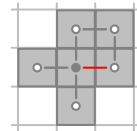
- **Postcondicions:**

*A\*0\*\* : es desconnecta del mòdul de l'Oest.*

<b>Regla:</b>	<i>Build scan tree Not-East</i>
<b>Prioritat:</b>	<i>10000</i>
<b>Precondicions:</b>	
<i>Sbst**</i>	
<i>F0,1</i>	
<i>F1,1</i>	
<b>Postcondicions:</b>	
<i>A**0*</i>	

- Regla: Build scan tree Not-East

*Aquesta regla fa referència als mòduls de la fase bst que han de desconnectar-se del seu Est.*



- Prioritat: 10000

*La prioritat és la mateixa que la de les fases [ini] i [bpf] per poder treballar en paral·lel. La prioritat és la mateixa que la resta de regles de la fase. Això permet aplicar aquesta regla i la de canvi d'estat simultàniament.*

- Precondicions:

*Sbst\*\* : afecta els mòduls de la fase [bst]*

*F0,1 : afecta els mòduls que tenen un veí al Nord.*

*F1,1 : afecta els mòduls que tenen un veí al Nord-Est.*

- Postcondicions:

*A\*\*0\* : es desconnecta del mòdul de l'Est.*

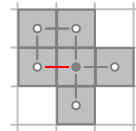
## Annex A - Regles de la fase de construcció de l'scan tree

---

<b>Regla:</b>	<i>Build scan tree Not-West</i>
<b>Prioritat:</b>	<i>10000</i>
<b>Precondicions:</b>	<i>Sbst** F0,1 F-1,1</i>
<b>Postcondicions:</b>	<i>A**0*</i>

- Regla: Build scan tree Not-West

*Aquesta regla fa referència als mòduls de la fase bst que han de desconnectar-se del seu Oest.*



- Prioritat: 10000

*La prioritat és la mateixa que la de les fases [ini] i [bpf] per poder treballar en paral·lel.*

*La prioritat és la mateixa que la resta de regles de la fase. Això permet aplicar aquesta regla i la de canvi d'estat simultàniament.*

- Precondicions:

*Sbst\*\* : afecta els mòduls de la fase [bst]*

*F0,1 : afecta els mòduls que tenen un veí al Nord.*

*F-1,1 : afecta els mòduls que tenen un veí al Nord-Oest.*

- Postcondicions:

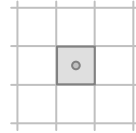
*A\*\*0\* : es desconnecta del mòdul de l'Oest.*



Regla:	Build scan tree Others
Prioritat:	10000
Precondicions:	
	Sbst**
Postcondicions:	
	Sbpf_*

- Regla: Build scan tree Others

Aquesta regla fa referència a tots els mòduls de la fase [bst], tant els que han realitzat desconnexions com els que s'han mantingut sense fer res. Tots canvien de fase.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bpf] per poder treballar en paral·lel. La prioritat és la mateixa que la resta de regles de la fase. Això permet aplicar aquesta regla simultàniament a les regles que realitzen les desconnexions.

- Precondicions:

Sbst\*\* : afecta els mòduls de la fase [bst]

- Postcondicions:

Sbpf\_\* : passen a la fase [bpf] (Build potential function) tot mantenint la informació de màster i de líder de forat.



## Annex B

### *Regles de la fase de construcció de la funció potencial*

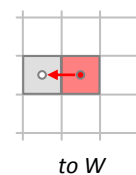
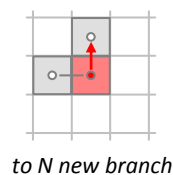
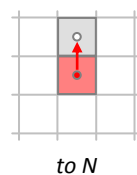
En aquest apartat es detallen les regles utilitzades en cada grup de la fase de construcció de la funció potencial [bpf] (*Build potential function*). L'objectiu, l'estratègia i una visió general d'aquest conjunt de regles es descriuen al Capítol 5.

#### B.1 Grup: "start"

Aquest grup està format per 3 regles que contemplen les tres formes possibles de que el màster iniciï el procés: que estigui connectat pel Nord, connectat per l'Oest, o per tots dos alhora. Les regles indiquen com ha d'actuar el màster en cada cas.

En aquest grup hem implementat les regles següents:

- Start potential function

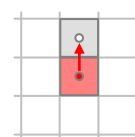


## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Start potential function to N</i>
<b>Prioritat:</b>	<i>10000</i>
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S****M</i> <i>S***_*</i> <i>A10**</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + 0000 0000</i> <i>C001 + 0000 0002</i> <i>C003 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Start potential function to N

Aquesta regla s'aplica al màster que inicia la fase de construcció de la funció potencial pel seu Nord, quan aquesta és l'única branca per la qual anar.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\*M* : afecta els mòduls identificats com a màster.

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*A10\*\** : afecta els mòduls que estan connectats al Nord però no a l'Oest.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + 0000 0000* : *Vpot = 0*

*C001 + 0000 0002* : *Signe = 2*

*C003 + 0000 0000* : *Min = 0*

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: (*Vpot = Vpot*).

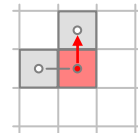
*#N02 + C003 0000* : missatge pel canal 2 al Nord: (*Min = Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Start potential function to N new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S****M</i> <i>S***_*</i> <i>A11**</i> <i>T0,1, bpf_*</i> <i>! T-1,0, ini**</i> <i>! T-1,0, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + 0000 0000</i> <i>C001 + 0000 0002</i> <i>C003 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Start potential function to N new branch

Aquesta regla s'aplica al màster que inicia la fase de construcció de la funció potencial pel seu Nord on, tenint en compte que hi ha dues branques, aquesta és la primera per la qual anar.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\*M* : afecta els mòduls identificats com a màster.

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*A11\*\** : afecta els mòduls que estan connectats al Nord i a l'Oest.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de *bpf\_\**.

*! T-1,0, ini\*\** : afecta els mòduls en què el seu veí de l'Oest ja no està en la fase [ini].

*! T-1,0, bst\*\** : afecta els mòduls en què el seu veí de l'Oest ja no està en la fase [bst].

- **Postcondicions:**

*C000 + 0000 0000* : *Vpot = 0*

*C001 + 0000 0002* : *Signe = 2*

*C003 + 0000 0000* : *Min = 0*

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: (*Vpot = Vpot*).

*#N02 + C000 0001* : missatge pel canal 2 al Nord: (*Min = Vpot+1*).

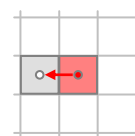
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Start potential function to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S****M</i> <i>S***_*</i> <i>A01**</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + 0000 0000</i> <i>C001 + 0000 0002</i> <i>C003 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Start potential function to W

Aquesta regla s'aplica al màster que inicia la fase de construcció de la funció potencial pel seu Oest, quan aquesta és l'única branca per la qual anar.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\*M* : afecta els mòduls identificats com a màster.

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*A01\*\** : afecta els mòduls que estan connectats a l'Oest però no al Nord.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + 0000 0000* : *Vpot* = 0

*C001 + 0000 0002* : *Signe* = 2

*C003 + 0000 0000* : *Min* = 0

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: (*Vpot* = *Vpot*).

*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: (*Min* = *Min*).

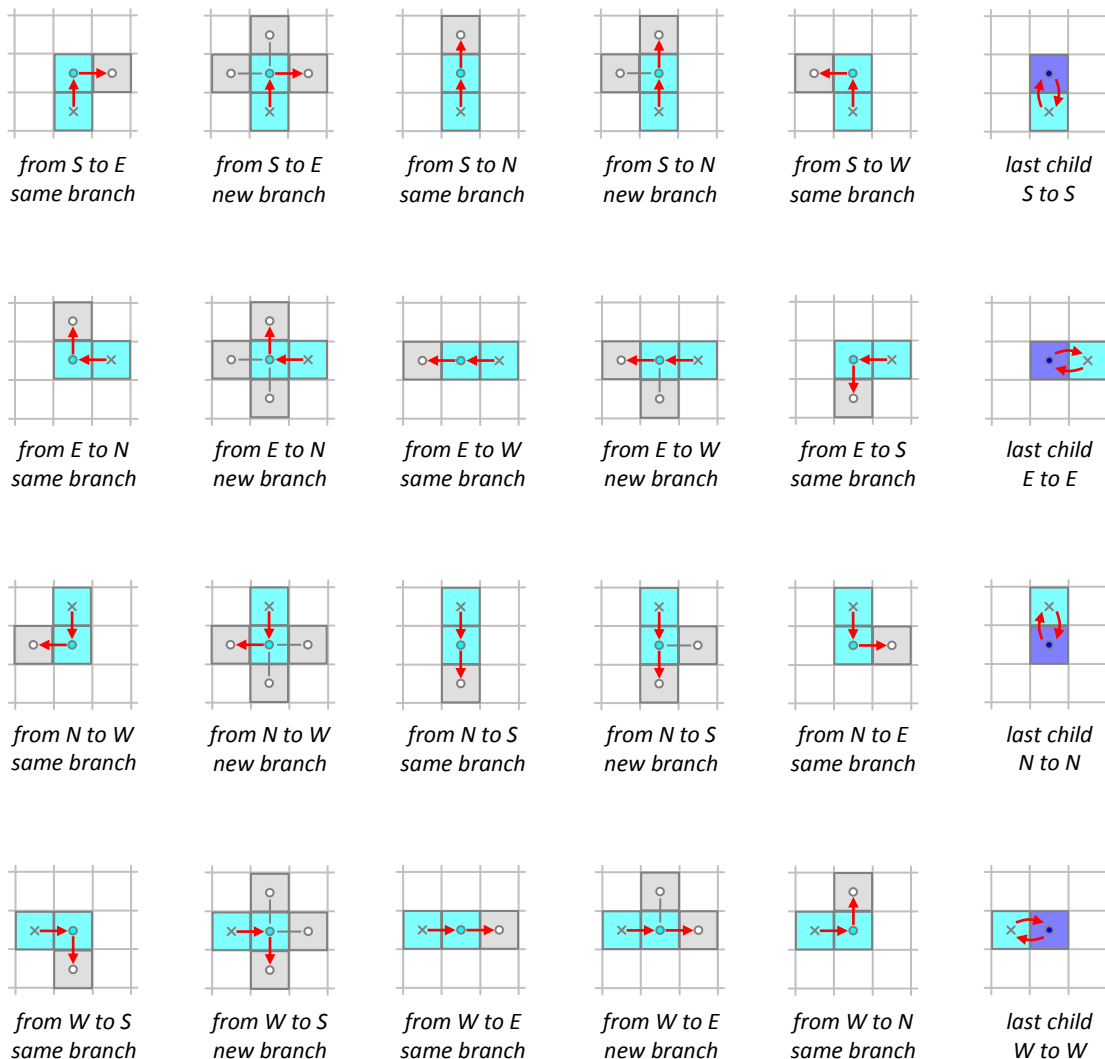
*S\*\*\*m\** : canvia el seu estat a "marcat".

## B.2 Grup: "first go"

Aquest grup està format per 24 regles subdividides en 4 subgrups de 6 regles. Cadascun dels subgrups fa referència a cadascuna de les 4 coordenades des d'on pot arribar el missatge de la funció potencial (N, S, E, W). Cadascun d'aquests subgrups té 3 regles de tipus "mateixa branca", i 3 regles de tipus "nova branca".

En aquest grup hem implementat les regles següents:

- Spread potential function first

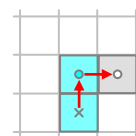


- Missatges des del Sud:

<b>Regla:</b>	<i>Spread potential function first from S to E same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#S05 0001</i> <i>A0011</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>S***m*</i>	

- Regla: Spread potential function first from S to E same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A0011* : afecta els mòduls que estan connectats només al Sud i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de bpf\_.

- Postcondicions:

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* :  $Signe = 2$

*C002 + 0000 0004* :  $Pare = Sud$

*C003 + #S02 0000* :  $Min = Min(S)$

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C003 0000* : missatge pel canal 2 a l'Est: ( $Min = Min$ ).

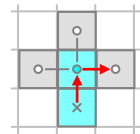
*S\*\*\*m\** : canvia el seu estat a "marcat".



<b>Regla:</b>	<i>Spread potential function first from S to E new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#S05 0001</i> <i>A**11</i> <i>T1,0, bpf_*</i> <i>!(</i> <i>!(A1011 !T0,1, ini** !T0,1, bst** )</i> <i>!(A0111 !T-1,0, ini** !T-1,0, bst** )</i> <i>!(A1111 !T0,1, ini** !T0,1, bst** !T-1,0, ini** !T-1,0, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to E new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar per l'Est, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A\*\*11* : afecta els mòduls que estan connectats al Sud i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *bpf\_\**.

*!( !A1011 !T0,1, ini\*\* !T0,1, bst\*\* ) !(A0111 !T-1,0, ini\*\* !T-1,0, bst\*\* ) !(A1111 !T0,1, ini\*\* !T0,1, bst\*\* !T-1,0, ini\*\* !T-1,0, bst\*\* ) )* : afecta els mòduls que estan connectats a altres mòduls a part de Sud i Est, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + #S02 0000* : *Min* = *Min*(S)

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C000 0001* : missatge pel canal 2 a l'Est: ( $Min = V_{pot}+1$ ).

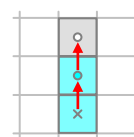
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from S to N same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#S05 0001</i> <i>A1001</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to N same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar pel Nord, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A1001* : afecta els mòduls que estan connectats només al Sud i al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

- **Postcondicions:**

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + #S02 0000* : *Min* = *Min*(S)

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

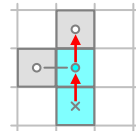
*#N02 + C003 0000* : missatge pel canal 2 al Nord: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from S to N new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#S05 0001</i> <i>A1101</i> <i>T0,1, bpf_*</i> <i>! T-1,0, ini**</i> <i>! T-1,0, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to N new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar pel Nord, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A1101* : afecta els mòduls que estan connectats només al Sud, al Nord i a l'Oest.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

*! T-1,0, ini\*\** : afecta els mòduls en què el seu veí de l'Oest ja no està en la fase [ini].

*! T-1,0, bst\*\** : afecta els mòduls en què el seu veí de l'Oest ja no està en la fase [bst].

- **Postcondicions:**

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + #S02 0000* : *Min* = *Min*(S)

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

*#N02 + C000 0001* : missatge pel canal 2 al Nord: ( $Min = V_{pot}+1$ ).

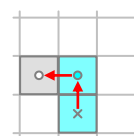
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from S to W same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#S05 0001</i> <i>A0101</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to W same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar per l'Oest, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A0101* : afecta els mòduls que estan connectats només al Sud i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + #S02 0000* : *Min* = *Min*(S)

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

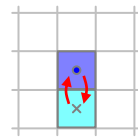
*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first last child S to S</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#S05 0001</i> <i>A0001</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>C004 + C000 0001</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child S to S

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A0001* : afecta els mòduls que estan connectats només al Sud.

- **Postcondicions:**

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + #S02 0000* : *Min* = *Min*(S)

*C004 + C000 0001* : *Max* =  $V_{pot} + 1$

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

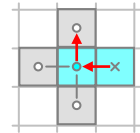
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].



<b>Regla:</b>	<i>Spread potential function first from E to N new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
$S_{bpf}^{**}$ $S^{***}_{**}$ $=\#E05\ 0001$ $A1^{*1*}$ $T0,1, bpf_{**}$ $!($ $!(A1110\ !T-1,0, ini^{**}\ !T-1,0, bst^{**})$ $!(A1011\ !T0,-1, ini^{**}\ !T0,-1, bst^{**})$ $!(A1111\ !T-1,0, ini^{**}\ !T-1,0, bst^{**}\ !T0,-1, ini^{**}\ !T0,-1, bst^{**})$ $)$	
<b>Postcondicions:</b>	
$C000 + \#E01\ 0001$ $C001 + 0000\ 0002$ $C002 + 0000\ 0003$ $C003 + \#E02\ 0000$ $\#N05 + 0000\ 0001$ $\#N01 + C000\ 0000$ $\#N02 + C000\ 0001$ $S^{***m*}$	

- **Regla:** Spread potential function first from E to N new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Est i l'han de propagar pel Nord, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

$S_{bpf}^{**}$  : afecta els mòduls de la fase [bpf]

$S^{***}_{**}$  : afecta els mòduls que no han estat marcats.

$=\#E05\ 0001$  : afecta els mòduls que reben un missatge de "marca" de l'Est.

$A1^{*1*}$  : afecta els mòduls que estan connectats a l'Est i al Nord.

$T0,1, bpf_{**}$  : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

$!(\ !A1110\ !T-1,0, ini^{**}\ !T-1,0, bst^{**})\ !A1011\ !T0,-1, ini^{**}\ !T0,-1, bst^{**})\ !A1111\ !T-1,0, ini^{**}\ !T-1,0, bst^{**}\ !T0,-1, ini^{**}\ !T0,-1, bst^{**})$  : afecta els mòduls que estan connectats a altres mòduls a part de Est i Nord, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

$C000 + \#E01\ 0001$  :  $V_{pot} = V_{pot}(E) + 1$

$C001 + 0000\ 0002$  :  $Signe = 2$

$C002 + 0000\ 0003$  :  $Pare = Est$

$C003 + \#E02\ 0000$  :  $Min = Min(E)$

$\#N05 + 0000\ 0001$  : missatge de "marca" al Nord.

$\#N01 + C000\ 0000$  : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

$\#N02 + C000\ 0001$  : missatge pel canal 2 al Nord: ( $Min = V_{pot}+1$ ).

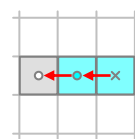
$S^{***m*}$  : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from E to W same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#E05 0001</i> <i>A0110</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #E01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + #E02 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to W same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Est i l'han de propagar per l'Oest, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*A0110* : afecta els mòduls que estan connectats només a l'Est i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + #E01 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0003* : *Pare* = Est

*C003 + #E02 0000* : *Min* = *Min*(E)

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: (*Min* = *Min*).

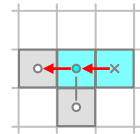
*S\*\*\*m\** : canvia el seu estat a "marcat".



<b>Regla:</b>	<i>Spread potential function first from E to W new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#E05 0001</i> <i>A0111</i> <i>T-1,0, bpf_*</i> <i>! T0,-1, ini**</i> <i>! T0,-1, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + #E01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + #E02 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to W new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Est i l'han de propagar per l'Oest, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*A0111* : afecta els mòduls que estan connectats només a l'Est, a l'Oest i al Sud.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

*! T0,-1, ini\*\** : afecta els mòduls en què el seu veí del Sud ja no està en la fase [ini].

*! T0,-1, bst\*\** : afecta els mòduls en què el seu veí del Sud ja no està en la fase [bst].

- **Postcondicions:**

*C000 + #E01 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0003* : *Pare* = Est

*C003 + #E02 0000* : *Min* = *Min*(E)

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

*#W02 + C000 0001* : missatge pel canal 2 a l'Oest: ( $Min = V_{pot}+1$ ).

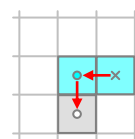
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from E to S same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#E05 0001</i> <i>A0011</i> <i>T0,-1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #E01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + #E02 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to S same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Est i l'han de propagar pel Sud, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*A0011* : afecta els mòduls que estan connectats només a l'Est i al Sud.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de bpf\_.

- **Postcondicions:**

*C000 + #E01 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0003* : *Pare* = Est

*C003 + #E02 0000* : *Min* = *Min*(E)

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

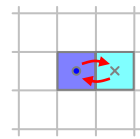
*#S02 + C003 0000* : missatge pel canal 2 al Sud: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first last child E to E</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#E05 0001</i> <i>A0010</i>	
<b>Postcondicions:</b>	
<i>C000 + #E01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + #E02 0000</i> <i>C004 + C000 0001</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child E to E

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Est i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*A0010* : afecta els mòduls que estan connectats només a l'Est.

- **Postcondicions:**

*C000 + #E01 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0003* : *Pare* = Est

*C003 + #E02 0000* : *Min* = *Min*(E)

*C004 + C000 0001* : *Max* =  $V_{pot} + 1$

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

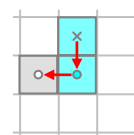
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

- Missatges des del Nord:

<b>Regla:</b>	<i>Spread potential function first from N to W same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#N05 0001</i> <i>A1100</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #N01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + #N02 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- Regla: Spread potential function first from N to W same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i l'han de propagar per l'Oest, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A1100* : afecta els mòduls que estan connectats només al Nord i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

- Postcondicions:

*C000 + #N01 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + #N02 0000* : *Min* = *Min*(N)

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

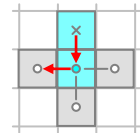
*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from N to W new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#N05 0001</i> <i>A11**</i> <i>T-1,0, bpf_*</i> <i>!(</i> <i>!(A1101 !T0,-1, ini** !T0,-1, bst** )</i> <i>!(A1110 !T1,0, ini** !T1,0, bst** )</i> <i>!(A1111 !T0,-1, ini** !T0,-1, bst** !T1,0, ini** !T1,0, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + #N01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + #N02 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to W new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i l'han de propagar per l'Oest, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A11\*\** : afecta els mòduls que estan connectats al Nord i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

*!( !A1101 !T0,-1, ini\*\* !T0,-1, bst\*\* ) !(A1110 !T1,0, ini\*\* !T1,0, bst\*\* ) !(A1111 !T0,-1, ini\*\* !T0,-1, bst\*\* !T1,0, ini\*\* !T1,0, bst\*\* ) )* : afecta els mòduls que estan connectats a altres mòduls a part de Nord i Oest, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + #N01 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + #N02 0000* : *Min* = *Min*(N)

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

*#W02 + C000 0001* : missatge pel canal 2 a l'Oest: ( $Min = V_{pot}+1$ ).

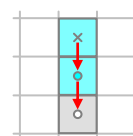
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from N to S same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#N05 0001</i> <i>A1001</i> <i>T0,-1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #N01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + #N02 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to S same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i l'han de propagar pel Sud, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A1001* : afecta els mòduls que estan connectats només al Sud i al Nord.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de bpf\_.

- **Postcondicions:**

*C000 + #N01 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + #N02 0000* : *Min* = *Min*(N)

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

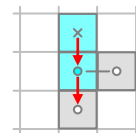
*#S02 + C003 0000* : missatge pel canal 2 al Sud: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from N to S new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#N05 0001</i> <i>A1011</i> <i>T0,-1, bpf_*</i> <i>! T1,0, ini**</i> <i>! T1,0, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + #N01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + #N02 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to S new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i l'han de propagar pel Sud, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A1011* : afecta els mòduls que estan connectats només al Nord, al Sud i a l'Est.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de bpf\_.

*! T1,0, ini\*\** : afecta els mòduls en què el seu veí de l'Est ja no està en la fase [ini].

*! T1,0, bst\*\** : afecta els mòduls en què el seu veí de l'Est ja no està en la fase [bst].

- **Postcondicions:**

*C000 + #N01 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + #N02 0000* : *Min* = *Min*(N)

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

*#S02 + C000 0001* : missatge pel canal 2 al Sud: ( $Min = V_{pot}+1$ ).

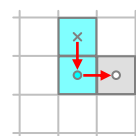
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from N to E same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#N05 0001</i> <i>A1010</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #N01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + #N02 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to E same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A1010* : afecta els mòduls que estan connectats només al Nord i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + #N01 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + #N02 0000* : *Min* = *Min*(N)

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C003 0000* : missatge pel canal 2 a l'Est: (*Min* = *Min*).

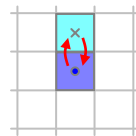
*S\*\*\*m\** : canvia el seu estat a "marcat".



<b>Regla:</b>	<i>Spread potential function first last child N to N</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#N05 0001</i> <i>A1000</i>	
<b>Postcondicions:</b>	
<i>C000 + #N01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + #N02 0000</i> <i>C004 + C000 0001</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child N to N

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A1000* : afecta els mòduls que estan connectats només al Nord.

- **Postcondicions:**

*C000 + #N01 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + #N02 0000* : *Min* = *Min*(N)

*C004 + C000 0001* : *Max* = *Vpot* + 1

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

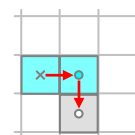
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

- Missatges des de l'Oest:

<b>Regla:</b>	<i>Spread potential function first from W to S same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** _ *</i> <i>=#W05 0001</i> <i>A0101</i> <i>T0,-1, bpf_ *</i>	
<b>Postcondicions:</b>	
<i>C000 + #W01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + #W02 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C003 0000</i> <i>S***m*</i>	

- Regla: Spread potential function first from W to S same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i l'han de propagar pel Sud, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \_ \** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A0101* : afecta els mòduls que estan connectats només a l'Oest i al Sud.

*T0,-1, bpf\_ \** : afecta els mòduls que tenen el veí del Sud en estat de bpf\_.

- Postcondicions:

*C000 + #W01 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = Oest

*C003 + #W02 0000* : *Min* = *Min*(W)

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

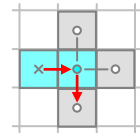
*#S02 + C003 0000* : missatge pel canal 2 al Sud: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from W to S new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#W05 0001</i> <i>A*1*1</i> <i>T0,-1, bpf_*</i> <i>!(</i> <i>!(A0111 !T1,0, ini** !T1,0, bst** )</i> <i>!(A1101 !T0,1, ini** !T0,1, bst** )</i> <i>!(A1111 !T1,0, ini** !T1,0, bst** !T0,1, ini** !T0,1, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + #W01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + #W02 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to S new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i l'han de propagar pel Sud, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A\*1\*1* : afecta els mòduls que estan connectats a l'Oest i al Sud.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de *bpf\_\**.

*!( !(A0111 !T1,0, ini\*\* !T1,0, bst\*\* ) !(A1101 !T0,1, ini\*\* !T0,1, bst\*\* ) !(A1111 !T1,0, ini\*\* !T1,0, bst\*\* !T0,1, ini\*\* !T0,1, bst\*\* ) )* : afecta els mòduls que estan connectats a altres mòduls a part d'Oest i Sud, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + #W01 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = Oest

*C003 + #W02 0000* : *Min* = *Min*(W)

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

*#S02 + C000 0001* : missatge pel canal 2 al Sud: ( $Min = V_{pot}+1$ ).

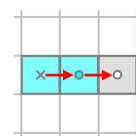
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from W to E same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#W05 0001</i> <i>A0110</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #W01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + #W02 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to E same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A0110* : afecta els mòduls que estan connectats només a l'Est i a l'Oest.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + #W01 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = *Oest*

*C003 + #W02 0000* : *Min* = *Min(W)*

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

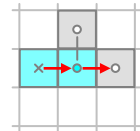
*#E02 + C003 0000* : missatge pel canal 2 a l'Est: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from W to E new branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#W05 0001</i> <i>A1110</i> <i>T1,0, bpf_*</i> <i>! T0,1, ini**</i> <i>! T0,1, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + #W01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + #W02 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to E new branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i l'han de propagar per l'Est, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A1110* : afecta els mòduls que estan connectats només a l'Oest, a l'Est i al Nord.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de bpf\_.

*! T0,1, ini\*\** : afecta els mòduls en què el seu veí del Nord ja no està en la fase [ini].

*! T0,1, bst\*\** : afecta els mòduls en què el seu veí del Nord ja no està en la fase [bst].

- **Postcondicions:**

*C000 + #W01 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = *Oest*

*C003 + #W02 0000* : *Min* = *Min(W)*

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C000 0001* : missatge pel canal 2 a l'Est: ( $Min = V_{pot}+1$ ).

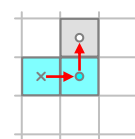
*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from W to N same branch</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#W05 0001</i> <i>A1100</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #W01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + #W02 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to N same branch

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i l'han de propagar pel Nord, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A1100* : afecta els mòduls que estan connectats només a l'Oest i al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

- **Postcondicions:**

*C000 + #W01 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = Oest

*C003 + #W02 0000* : *Min* = *Min*(W)

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

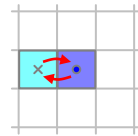
*#N02 + C003 0000* : missatge pel canal 2 al Nord: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first last child W to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***_*</i> <i>=#W05 0001</i> <i>A0100</i>	
<b>Postcondicions:</b>	
<i>C000 + #W01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + #W02 0000</i> <i>C004 + C000 0001</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child W to W

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*\_\** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

- **Postcondicions:**

*C000 + #W01 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = Oest

*C003 + #W02 0000* : *Min* = *Min*(W)

*C004 + C000 0001* : *Max* = *Vpot* + 1

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

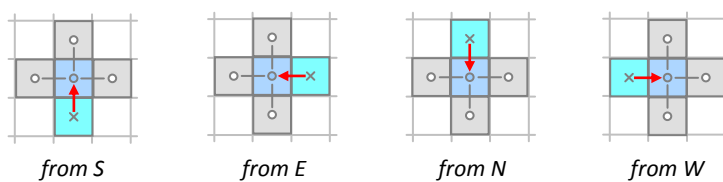
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

### **B.3 Grup: “first wait”**

Aquest grup està format per 4 regles. Cadascuna contempla un dels 4 punts cardinals per on pot arribar el missatge de funció potencial (N, S, E, W).

En aquest grup hem implementat les regles següents:

- Spread potential function first wait

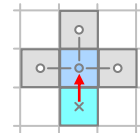




<b>Regla:</b>	<i>Spread potential function first from S wait</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#S05 0001</i> <i>A***1</i> <i>!(A0011 T1,0, bpf_*)</i> <i>!(A**11 T1,0, bpf_* !(A1011 !T0,1, ini** !T0,1, bst**) !(A0111 !T-1,0, ini** !T-1,0, bst**)</i> <i>!(A1111 !T0,1, ini** !T0,1, bst** !T-1,0, ini** !T-1,0, bst**))</i> <i>!(A1001 T0,1, bpf_*)</i> <i>!(A1101 T0,1, bpf_* !T-1,0, ini** !T-1,0, bst**)</i> <i>!(A0101 T-1,0, bpf_*)</i> <i>!(A0001)</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0004</i> <i>C011 + #S01 0000</i> <i>C012 + #S02 0000</i> <i>S***w*</i>	

- **Regla:** Spread potential function first from S wait

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i no han pogut propagar el missatge al mòdul següent.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A\*\*\*1* : afecta els mòduls que estan connectats al Sud.

*!(A0011 T1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from S to E same branch".

*!(A\*\*11 T1,0, bpf\_\* !(A1011 !T0,1, ini\*\* !T0,1, bst\*\* ) !(A0111 !T-1,0, ini\*\* !T-1,0, bst\*\* ) !(A1111 !T0,1, ini\*\* !T0,1, bst\*\* !T-1,0, ini\*\* !T-1,0, bst\*\* ))* : afecta els mòduls que no apliquen la regla "Spread potential function first from S to E new branch".

*!(A1001 T0,1, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from S to N same branch".

*!(A1101 T0,1, bpf\_\* !T-1,0, ini\*\* !T-1,0, bst\*\* )* : afecta els mòduls que no apliquen la regla "Spread potential function first from S to N new branch".

*!(A0101 T-1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from S to W same branch".

*!(A0001)* : afecta els mòduls que no apliquen la regla "Spread potential function first last child S to S".

- **Postcondicions:**

*C010 + 0000 0004* : emmagatzema la direcció del missatge = Sud

*C011 + #S01 0000* : emmagatzema el missatge del canal 1 = Vpot(S)

*C012 + #S02 0000* : emmagatzema el missatge del canal 2 = Min(S)

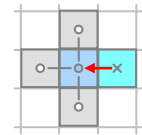
*S\*\*\*w\** : canvia el seu estat a "en espera".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from E wait</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#E05 0001</i> <i>A**1*</i> <i>!(A1010 T0,1, bpf_*)</i> <i>!(A1*1* T0,1, bpf_* !( !(A1110 !T-1,0, ini** !T-1,0, bst** ) !(A1011 !T0,-1, ini** !T0,-1, bst** )</i> <i>!(A1111 !T-1,0, ini** !T-1,0, bst** !T0,-1, ini** !T0,-1, bst** )))</i> <i>!(A0110 T-1,0, bpf_*)</i> <i>!(A0111 T-1,0, bpf_* !T0,-1, ini** !T0,-1, bst** )</i> <i>!(A0011 T0,-1, bpf_*)</i> <i>!(A0010 )</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0003</i> <i>C011 + #E01 0000</i> <i>C012 + #E02 0000</i> <i>S***w*</i>	

- **Regla:** Spread potential function first from E wait

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Est i no han pogut propagar el missatge al mòdul següent.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*A\*\*1\** : afecta els mòduls que estan connectats a l'Est.

*!(A1010 T0,1, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from E to N same branch".

*!(A1\*1\* T0,1, bpf\_\* !( !(A1110 !T-1,0, ini\*\* !T-1,0, bst\*\* ) !(A1011 !T0,-1, ini\*\* !T0,-1, bst\*\* ) !(A1111 !T-1,0, ini\*\* !T-1,0, bst\*\* !T0,-1, ini\*\* !T0,-1, bst\*\* )))* : afecta els mòduls que no apliquen la regla "Spread potential function first from E to N new branch".

*!(A0110 T-1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from E to W same branch".

*!(A0111 T-1,0, bpf\_\* !T0,-1, ini\*\* !T0,-1, bst\*\* )* : afecta els mòduls que no apliquen la regla "Spread potential function first from E to W new branch".

*!(A0011 T0,-1, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from E to S same branch".

*!(A0010 )* : afecta els mòduls que no apliquen la regla "Spread potential function first last child E to E".

- **Postcondicions:**

*C010 + 0000 0003* : emmagatzema la direcció del missatge = Est

*C011 + #E01 0000* : emmagatzema el missatge del canal 1 = Vpot(E)

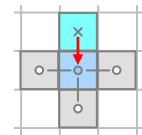
*C012 + #E02 0000* : emmagatzema el missatge del canal 2 = Min(E)

*S\*\*\*w\** : canvia el seu estat a "en espera".

<b>Regla:</b>	<i>Spread potential function first from N wait</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#N05 0001</i> <i>A1***</i> <i>!(A1100 T-1,0, bpf_*)</i> <i>!(A11** T-1,0, bpf_* !(A1101 !T0,-1, ini** !T0,-1, bst**) !(A1110 !T1,0, ini** !T1,0, bst** )</i> <i>!(A1111 !T0,-1, ini** !T0,-1, bst** !T1,0, ini** !T1,0, bst** )))</i> <i>!(A1001 T0,-1, bpf_*)</i> <i>!(A1011 T0,-1, bpf_* !T1,0, ini** !T1,0, bst** )</i> <i>!(A1010 T1,0, bpf_*)</i> <i>!(A1000 )</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0001</i> <i>C011 + #N01 0000</i> <i>C012 + #N02 0000</i> <i>S***w*</i>	

- **Regla:** Spread potential function first from N wait

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Nord i no han pogut propagar el missatge al mòdul següent.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A1\*\*\** : afecta els mòduls que estan connectats al Nord.

*!(A1100 T-1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from N to W same branch".

*!(A11\*\* T-1,0, bpf\_\* !(A1101 !T0,-1, ini\*\* !T0,-1, bst\*\*) !(A1110 !T1,0, ini\*\* !T1,0, bst\*\* ) !(A1111 !T0,-1, ini\*\* !T0,-1, bst\*\* !T1,0, ini\*\* !T1,0, bst\*\* )))* : afecta els mòduls que no apliquen la regla "Spread potential function first from N to W new branch".

*!(A1001 T0,-1, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from N to S same branch".

*!(A1011 T0,-1, bpf\_\* !T1,0, ini\*\* !T1,0, bst\*\* )* : afecta els mòduls que no apliquen la regla "Spread potential function first from N to S new branch".

*!(A1010 T1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from N to E same branch".

*!(A1000 )* : afecta els mòduls que no apliquen la regla "Spread potential function first last child N to N".

- **Postcondicions:**

*C010 + 0000 0001* : emmagatzema la direcció del missatge = Nord

*C011 + #N01 0000* : emmagatzema el missatge del canal 1 =  $V_{pot}(N)$

*C012 + #N02 0000* : emmagatzema el missatge del canal 2 =  $Min(N)$

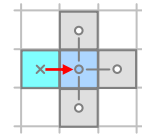
*S\*\*\*w\** : canvia el seu estat a "en espera".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from W wait</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S*** *</i> <i>=#W05 0001</i> <i>A*1**</i> <i>!(A0101 T0,-1, bpf_*)</i> <i>!(A*1*1 T0,-1, bpf_* !(A0111 !T1,0, ini** !T1,0, bst**) !(A1101 !T0,1, ini** !T0,1, bst**) !(A1111 !T1,0, ini** !T1,0, bst** !T0,1, ini** !T0,1, bst**)))</i> <i>!(A0110 T1,0, bpf_*)</i> <i>!(A1110 T1,0, bpf_* !T0,1, ini** !T0,1, bst**)</i> <i>!(A1100 T0,1, bpf_*)</i> <i>!(A0100)</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0002</i> <i>C011 + #W01 0000</i> <i>C012 + #W02 0000</i> <i>S***w*</i>	

- **Regla:** Spread potential function first from W wait

Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) per l'Oest i no han pogut propagar el missatge al mòdul següent.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\* \** : afecta els mòduls que no han estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A\*1\*\** : afecta els mòduls que estan connectats a l'Oest.

*!(A0101 T0,-1, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from W to S same branch".

*!(A\*1\*1 T0,-1, bpf\_\* !(A0111 !T1,0, ini\*\* !T1,0, bst\*\*) !(A1101 !T0,1, ini\*\* !T0,1, bst\*\*) !(A1111 !T1,0, ini\*\* !T1,0, bst\*\* !T0,1, ini\*\* !T0,1, bst\*\*)))* : afecta els mòduls que no apliquen la regla "Spread potential function first from W to S new branch".

*!(A0110 T1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from W to E same branch".

*!(A1110 T1,0, bpf\_\* !T0,1, ini\*\* !T0,1, bst\*\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from W to E new branch".

*!(A1100 T0,1, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function first from W to N same branch".

*!(A0100)* : afecta els mòduls que no apliquen la regla "Spread potential function first last child W to W".

- **Postcondicions:**

*C010 + 0000 0002* : emmagatzema la direcció del missatge = Oest

*C011 + #W01 0000* : emmagatzema el missatge del canal 1 = Vpot(W)

*C012 + #W02 0000* : emmagatzema el missatge del canal 2 = Min(W)

*S\*\*\*w\** : canvia el seu estat a "en espera".

## B.4 Grup: "wait-continue"

Aquest grup està format per 24 regles subdividides en 4 subgrups de 6 regles. Cadascun dels subgrups fa referència a una de les 4 coordenades des d'on li va arribar el missatge que té emmagatzemat (N, S, E, W). Cadascun d'aquests subgrups té 3 regles de tipus "mateixa branca", i 3 regles de tipus "nova branca".

En aquest grup hem implementat les regles següents:

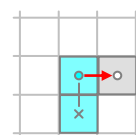
- Spread potential function first cont



- Missatges des del Sud:

<b>Regla:</b>	<i>Spread potential function first from S to E same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0004</i> <i>A0011</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to E same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Sud i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0004* : afecta els mòduls que van rebre el missatge del Sud.

*A0011* : afecta els mòduls que estan connectats només al Sud i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + C012 0000* : *Min* = *Min*(S)

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

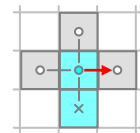
*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C003 0000* : missatge pel canal 2 a l'Est: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from S to E new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0004</i> <i>A**11</i> <i>T1,0, bpf_*</i> <i>!(</i> <i>!(A1011 !T0,1, ini** !T0,1, bst** )</i> <i>!(A0111 !T-1,0, ini** !T-1,0, bst** )</i> <i>!(A1111 !T0,1, ini** !T0,1, bst** !T-1,0, ini** !T-1,0, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to E new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Sud i l'han de propagar per l'Est, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0004* : afecta els mòduls que van rebre el missatge del Sud.

*A\*\*11* : afecta els mòduls que estan connectats al Sud i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de bpf\_.

*!( ! (A1011 !T0,1, ini\*\* !T0,1, bst\*\* ) ! (A0111 !T-1,0, ini\*\* !T-1,0, bst\*\* ) ! (A1111 !T0,1, ini\*\* !T0,1, bst\*\* !T-1,0, ini\*\* !T-1,0, bst\*\* ) )* : afecta els mòduls que estan connectats a altres mòduls a part de Sud i Est, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + C012 0000* : *Min* = *Min*(S)

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

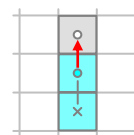
*#E02 + C000 0001* : missatge pel canal 2 a l'Est: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from S to N same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0004</i> <i>A1001</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to N same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Sud i l'han de propagar pel Nord, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0004* : afecta els mòduls que van rebre el missatge del Sud.

*A1001* : afecta els mòduls que estan connectats només al Sud i al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + C012 0000* : *Min* = *Min*(S)

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

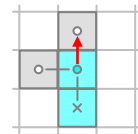
*#N02 + C003 0000* : missatge pel canal 2 al Nord: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".



<b>Regla:</b>	<i>Spread potential function first from S to N new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0004</i> <i>A1101</i> <i>T0,1, bpf_*</i> <i>!T-1,0, ini**</i> <i>!T-1,0, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to N new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Sud i l'han de propagar pel Nord, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0004* : afecta els mòduls que van rebre el missatge del Sud.

*A1101* : afecta els mòduls que estan connectats només al Sud, al Nord i a l'Oest.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

*!T-1,0, ini\*\** : afecta els mòduls en què el seu veí de l'Oest ja no està en la fase [ini].

*!T-1,0, bst\*\** : afecta els mòduls en què el seu veí de l'Oest ja no està en la fase [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe = 2*

*C002 + 0000 0004* : *Pare = Sud*

*C003 + C012 0000* : *Min = Min(S)*

*C010 + 0000 0000* : *neteja del comptador.*

*C011 + 0000 0000* : *neteja del comptador.*

*C012 + 0000 0000* : *neteja del comptador.*

*#N05 + 0000 0001* : *missatge de "marca" al Nord.*

*#N01 + C000 0000* : *missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).*

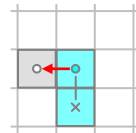
*#N02 + C000 0001* : *missatge pel canal 2 al Nord: ( $Min = V_{pot}+1$ ).*

*S\*\*\*m\** : *canvia el seu estat a "marcat".*

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from S to W same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0004</i> <i>A0101</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from S to W same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Sud i l'han de propagar per l'Oest, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0004* : afecta els mòduls que van rebre el missatge del Sud.

*A0101* : afecta els mòduls que estan connectats només al Sud i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de bpf\_.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* :  $Signe = 2$

*C002 + 0000 0004* :  $Pare = Sud$

*C003 + C012 0000* :  $Min = Min(S)$

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

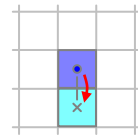
*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first last child S to S cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0004</i> <i>A0001</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + C012 0000</i> <i>C004 + C000 0001</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child S to S cont

Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Sud i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0004* : afecta els mòduls que van rebre el missatge del Sud.

*A0001* : afecta els mòduls que estan connectats només al Sud.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + C012 0000* : *Min* = *Min*(S)

*C004 + C000 0001* : *Max* =  $V_{pot} + 1$

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#S05 + 0000 0001* : missatge de "marca" al Sud.

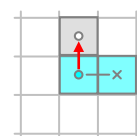
*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

- Missatges des de l'Est:

<b>Regla:</b>	<i>Spread potential function first from E to N same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0003</i> <i>A1010</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to N same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Est i l'han de propagar pel Nord, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0003* : afecta els mòduls que van rebre el missatge de l'Est.

*A1010* : afecta els mòduls que estan connectats només a l'Est i al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* :  $Signe = 2$

*C002 + 0000 0003* :  $Pare = Est$

*C003 + C012 0000* :  $Min = Min(E)$

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#N05 + 0000 0001* : missatge de "marca" al Nord.

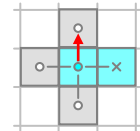
*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

*#N02 + C003 0000* : missatge pel canal 2 al Nord: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from E to N new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0003</i> <i>A1*1*</i> <i>T0,1, bpf_*</i> <i>!(</i> <i>!(A1110 !T-1,0, ini** !T-1,0, bst** )</i> <i>!(A1011 !T0,-1, ini** !T0,-1, bst** )</i> <i>!(A1111 !T-1,0, ini** !T-1,0, bst** !T0,-1, ini** !T0,-1, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to N new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Est i l'han de propagar pel Nord, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0003* : afecta els mòduls que van rebre el missatge de l'Est.

*A1\*1\** : afecta els mòduls que estan connectats a l'Est i al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

*!( ! (A1110 !T-1,0, ini\*\* !T-1,0, bst\*\* ) ! (A1011 !T0,-1, ini\*\* !T0,-1, bst\*\* ) ! (A1111 !T-1,0, ini\*\* !T-1,0, bst\*\* !T0,-1, ini\*\* !T0,-1, bst\*\* ) )* : afecta els mòduls que estan connectats a altres mòduls a part de Est i Nord, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* :  $Signe = 2$

*C002 + 0000 0003* :  $Pare = Est$

*C003 + C012 0000* :  $Min = Min(E)$

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

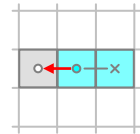
*#N02 + C000 0001* : missatge pel canal 2 al Nord: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from E to W same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0003</i> <i>A0110</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to W same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Est i l'han de propagar per l'Oest, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0003* : afecta els mòduls que van rebre el missatge de l'Est.

*A0110* : afecta els mòduls que estan connectats només a l'Est i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de bpf\_.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* :  $Signe = 2$

*C002 + 0000 0003* :  $Pare = Est$

*C003 + C012 0000* :  $Min = Min(E)$

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

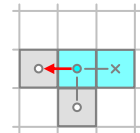
*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from E to W new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0003</i> <i>A0111</i> <i>T-1,0, bpf_*</i> <i>!T0,-1, ini**</i> <i>!T0,-1, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to W new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Est i l'han de propagar per l'Oest, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0003* : afecta els mòduls que van rebre el missatge de l'Est.

*A0111* : afecta els mòduls que estan connectats només a l'Est, a l'Oest i al Sud.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

*!T0,-1, ini\*\** : afecta els mòduls en què el seu veí del Sud ja no està en la fase [ini].

*!T0,-1, bst\*\** : afecta els mòduls en què el seu veí del Sud ja no està en la fase [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe = 2*

*C002 + 0000 0003* : *Pare = Est*

*C003 + C012 0000* : *Min = Min(E)*

*C010 + 0000 0000* : *neteja del comptador.*

*C011 + 0000 0000* : *neteja del comptador.*

*C012 + 0000 0000* : *neteja del comptador.*

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

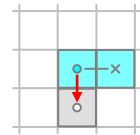
*#W02 + C000 0001* : missatge pel canal 2 a l'Oest: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from E to S same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0003</i> <i>A0011</i> <i>T0,-1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from E to S same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Est i l'han de propagar pel Sud, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0003* : afecta els mòduls que van rebre el missatge de l'Est.

*A0011* : afecta els mòduls que estan connectats només a l'Est i al Sud.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de *bpf\_*.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0003* : *Pare* = Est

*C003 + C012 0000* : *Min* = *Min(E)*

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

*#S02 + C003 0000* : missatge pel canal 2 al Sud: ( $Min = Min$ ).

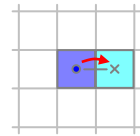
*S\*\*\*m\** : canvia el seu estat a "marcat".



<b>Regla:</b>	<i>Spread potential function first last child E to E cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0003</i> <i>A0010</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0003</i> <i>C003 + C012 0000</i> <i>C004 + C000 0001</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child E to E cont

Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Est i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0003* : afecta els mòduls que van rebre el missatge de l'Est.

*A0010* : afecta els mòduls que estan connectats només a l'Est.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0003* : *Pare* = Est

*C003 + C012 0000* : *Min* = *Min*(E)

*C004 + C000 0001* : *Max* =  $V_{pot} + 1$

*C010 + 0000 0000* : *neteja del comptador*.

*C011 + 0000 0000* : *neteja del comptador*.

*C012 + 0000 0000* : *neteja del comptador*.

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

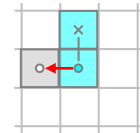
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

- Missatges des del Nord:

<b>Regla:</b>	<i>Spread potential function first from N to W same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0001</i> <i>A1100</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C003 0000</i> <i>S***m*</i>	

- Regla: Spread potential function first from N to W same branch cont

Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Nord i l'han de propagar per l'Oest, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0001* : afecta els mòduls que van rebre el missatge del Nord.

*A1100* : afecta els mòduls que estan connectats només al Nord i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

- Postcondicions:

*C000 + C011 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + C012 0000* : *Min* = *Min*(N)

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

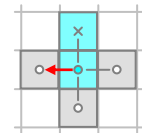
*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

*#W02 + C003 0000* : missatge pel canal 2 a l'Oest: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from N to W new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0001</i> <i>A11**</i> <i>T-1,0, bpf_*</i> <i>!(</i> <i>!(A1101 !T0,-1, ini** !T0,-1, bst** )</i> <i>!(A1110 !T1,0, ini** !T1,0, bst** )</i> <i>!(A1111 !T0,-1, ini** !T0,-1, bst** !T1,0, ini** !T1,0, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>#W02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to W new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Nord i l'han de propagar per l'Oest, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0001* : afecta els mòduls que van rebre el missatge del Nord.

*A11\*\** : afecta els mòduls que estan connectats al Nord i a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

*!( ! (A1101 !T0,-1, ini\*\* !T0,-1, bst\*\* ) ! (A1110 !T1,0, ini\*\* !T1,0, bst\*\* ) ! (A1111 !T0,-1, ini\*\* !T0,-1, bst\*\* !T1,0, ini\*\* !T1,0, bst\*\* ) )* : afecta els mòduls que estan connectats a altres mòduls a part de Nord i Oest, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + C012 0000* : *Min* = *Min(N)*

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

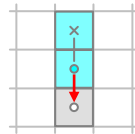
*#W02 + C000 0001* : missatge pel canal 2 a l'Oest: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from N to S same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0001</i> <i>A1001</i> <i>T0,-1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to S same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Nord i l'han de propagar pel Sud, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0001* : afecta els mòduls que van rebre el missatge del Nord.

*A1001* : afecta els mòduls que estan connectats només al Sud i al Nord.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de *bpf\_*.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + C012 0000* : *Min* = *Min(N)*

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#S05 + 0000 0001* : missatge de "marca" al Sud.

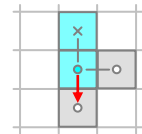
*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

*#S02 + C003 0000* : missatge pel canal 2 al Sud: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from N to S new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0001</i> <i>A1011</i> <i>T0,-1, bpf_*</i> <i>!T1,0, ini**</i> <i>!T1,0, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to S new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Nord i l'han de propagar pel Sud, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0001* : afecta els mòduls que van rebre el missatge del Nord.

*A1011* : afecta els mòduls que estan connectats només al Nord, al Sud i a l'Est.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de *bpf\_\**.

*!T1,0, ini\*\** : afecta els mòduls en què el seu veí de l'Est ja no està en la fase [ini].

*!T1,0, bst\*\** : afecta els mòduls en què el seu veí de l'Est ja no està en la fase [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + C012 0000* : *Min* = *Min(N)*

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

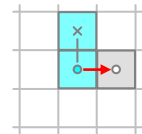
*#S02 + C000 0001* : missatge pel canal 2 al Sud: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from N to E same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0001</i> <i>A1010</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from N to E same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Nord i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0001* : afecta els mòduls que van rebre el missatge del Nord.

*A1010* : afecta els mòduls que estan connectats només al Nord i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de bpf\_.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + C012 0000* : *Min* = *Min*(N)

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

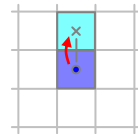
*#E02 + C003 0000* : missatge pel canal 2 a l'Est: (*Min* = *Min*).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first last child N to N cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0001</i> <i>A1000</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0001</i> <i>C003 + C012 0000</i> <i>C004 + C000 0001</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child N to N cont

Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial pel Nord i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0001* : afecta els mòduls que van rebre el missatge del Nord.

*A1000* : afecta els mòduls que estan connectats només al Nord.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0001* : *Pare* = Nord

*C003 + C012 0000* : *Min* = *Min*(N)

*C004 + C000 0001* : *Max* = *Vpot* + 1

*C010 + 0000 0000* : *neteja del comptador*.

*C011 + 0000 0000* : *neteja del comptador*.

*C012 + 0000 0000* : *neteja del comptador*.

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

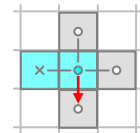
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].





<b>Regla:</b>	<i>Spread potential function first from W to S new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0002</i> <i>A*1*1</i> <i>T0,-1, bpf_*</i> <i>!(</i> <i>!(A0111 !T1,0, ini** !T1,0, bst** )</i> <i>!(A1101 !T0,1, ini** !T0,1, bst** )</i> <i>!(A1111 !T1,0, ini** !T1,0, bst** !T0,1, ini** !T0,1, bst** )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#S05 + 0000 0001</i> <i>#S01 + C000 0000</i> <i>#S02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to S new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Oest i l'han de propagar pel Sud, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0002* : afecta els mòduls que van rebre el missatge de l'Oest.

*A\*1\*1* : afecta els mòduls que estan connectats a l'Oest i al Sud.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de *bpf\_\**.

*!( !A0111 !T1,0, ini\*\* !T1,0, bst\*\* ) !(A1101 !T0,1, ini\*\* !T0,1, bst\*\* ) !(A1111 !T1,0, ini\*\* !T1,0, bst\*\* !T0,1, ini\*\* !T0,1, bst\*\* )* : afecta els mòduls que estan connectats a altres mòduls a part d'Oest i Sud, i tots ells han superat ja les fases de [ini] i [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = Oest

*C003 + C012 0000* : *Min* = *Min*(W)

*C010 + 0000 0000* : neteja del comptador.

*C011 + 0000 0000* : neteja del comptador.

*C012 + 0000 0000* : neteja del comptador.

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + C000 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}$ ).

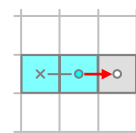
*#S02 + C000 0001* : missatge pel canal 2 al Sud: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from W to E same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0002</i> <i>A0110</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to E same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Oest i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0002* : afecta els mòduls que van rebre el missatge de l'Oest.

*A0110* : afecta els mòduls que estan connectats només a l'Est i a l'Oest.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de bpf\_.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = *Oest*

*C003 + C012 0000* : *Min* = *Min(W)*

*C010 + 0000 0000* : *neteja del comptador*.

*C011 + 0000 0000* : *neteja del comptador*.

*C012 + 0000 0000* : *neteja del comptador*.

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

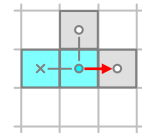
*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C003 0000* : missatge pel canal 2 a l'Est: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first from W to E new branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0002</i> <i>A1110</i> <i>T1,0, bpf_*</i> <i>!T0,1, ini**</i> <i>!T0,1, bst**</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C000 0001</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to E new branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Oest i l'han de propagar per l'Est, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0002* : afecta els mòduls que van rebre el missatge de l'Oest.

*A1110* : afecta els mòduls que estan connectats només a l'Oest, a l'Est i al Nord.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *bpf\_\**.

*!T0,1, ini\*\** : afecta els mòduls en què el seu veí del Nord ja no està en la fase [ini].

*!T0,1, bst\*\** : afecta els mòduls en què el seu veí del Nord ja no està en la fase [bst].

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe = 2*

*C002 + 0000 0002* : *Pare = Oest*

*C003 + C012 0000* : *Min = Min(W)*

*C010 + 0000 0000* : *neteja del comptador.*

*C011 + 0000 0000* : *neteja del comptador.*

*C012 + 0000 0000* : *neteja del comptador.*

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

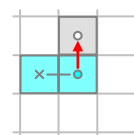
*#E02 + C000 0001* : missatge pel canal 2 a l'Est: ( $Min = V_{pot}+1$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from W to N same branch cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0002</i> <i>A1100</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + C012 0000</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#N05 + 0000 0001</i> <i>#N01 + C000 0000</i> <i>#N02 + C003 0000</i> <i>S***m*</i>	

- **Regla:** Spread potential function first from W to N same branch cont  
Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Oest i l'han de propagar pel Nord, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0002* : afecta els mòduls que van rebre el missatge de l'Oest.

*A1100* : afecta els mòduls que estan connectats només a l'Oest i al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de *bpf\_\**.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = *Oest*

*C003 + C012 0000* : *Min* = *Min(W)*

*C010 + 0000 0000* : *neteja del comptador*.

*C011 + 0000 0000* : *neteja del comptador*.

*C012 + 0000 0000* : *neteja del comptador*.

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + C000 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}$ ).

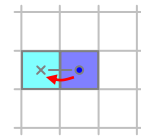
*#N02 + C003 0000* : missatge pel canal 2 al Nord: ( $Min = Min$ ).

*S\*\*\*m\** : canvia el seu estat a "marcat".

<b>Regla:</b>	<i>Spread potential function first last child W to W cont</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***w*</i> <i>=C010 0002</i> <i>A0100</i>	
<b>Postcondicions:</b>	
<i>C000 + C011 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0002</i> <i>C003 + C012 0000</i> <i>C004 + C000 0001</i> <i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>#W05 + 0000 0001</i> <i>#W01 + C000 0000</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function first last child W to W cont

Aquesta regla s'aplica als mòduls que es troben en espera, als quals els va arribar el missatge de funció potencial per l'Oest i l'han de retornar per la mateixa direcció perquè ha arribat al final de la branca. Aquests mòduls només estan connectats per una única posició.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*w\** : afecta els mòduls en estat d'espera.

*=C010 0002* : afecta els mòduls que van rebre el missatge de l'Oest.

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

- **Postcondicions:**

*C000 + C011 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0002* : *Pare* = *Oest*

*C003 + C012 0000* : *Min* = *Min*(*W*)

*C004 + C000 0001* : *Max* =  $V_{pot} + 1$

*C010 + 0000 0000* : *neteja del comptador*.

*C011 + 0000 0000* : *neteja del comptador*.

*C012 + 0000 0000* : *neteja del comptador*.

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + C000 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}$ ).

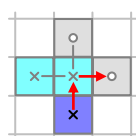
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

## **B.5 Grup: “continue”**

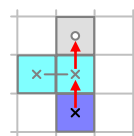
Aquest grup està format per 8 regles subdividides en 4 subgrups de 2 regles. Cada subgrup fa referència a cadascuna de les 4 coordenades des d'on pot arribar el missatge de la funció potencial (N, S, E, W). En cada cas, tracta els dos veïns possibles per on continuar amb la propagació.

En aquest grup hem implementat les regles següents:

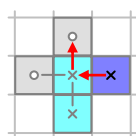
- Spread potential function continue



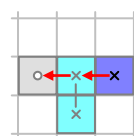
*from S to E*



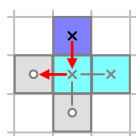
*from S to N*



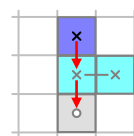
*from E to N*



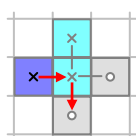
*from E to W*



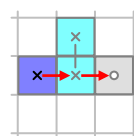
*from N to W*



*from N to S*



*from W to S*

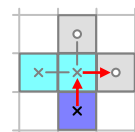


*from W to E*

<b>Regla:</b>	<i>Spread potential function continue from S to E</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
$S_{bpf}^{**}$ $S^{***}m^{*}$ $=\#S05\ 0001$ $A^{**}1^{*}$ $T1,0, bpf_{-}^{*}$	
<b>Postcondicions:</b>	
$\#E05 + 0000\ 0001$ $\#E01 + \#S01\ 0000$ $\#E02 + \#S01\ 0001$	

*Aquesta regla s'aplica als mòduls als quals arriba (de*

*funció potencial, aquest cop pel Sud i han de propagar-lo per l'Est, que forma part d'una altra branca perquè es troba en una bifurcació amb diversos mòduls connectats.*



*La prioritat és la m*

Sbpf\*\* : *afecta e*

$S^{***}m^*$  : afecta els mòduls que estat mar

=#S05 0001 : *afecta els mòduls que reben un m*

**A\*\*1\*** : *afecta els mòduls que estan connectats a l'Est.*

T1,0, bpf\_\* : *afecta els mòduls que tenen el veí de l'Est*

#E05 + 0000 0001

#E01 + #S01 0000 : *missatge pel canal 1 a l'Est:*

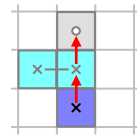
#E02 + #S01 0001 : *missatge pel canal 2 a l'Est: (Min = Vpot(S)+1*

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function continue from S to N</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#S05 0001</i> <i>A1***</i> <i>T0,1, bpf_*</i> <i>!(A**1* T1,0, bpf_*)</i>	
<b>Postcondicions:</b>	
<i>#N05 + 0000 0001</i> <i>#N01 + #S01 0000</i> <i>#N02 + #S01 0001</i>	

- **Regla:** Spread potential function continue from S to N

Aquesta regla s'aplica als mòduls als quals arriba (de nou) el missatge de funció potencial, aquest cop pel Sud i han de propagar-lo pel Nord, que és l'única branca que queda sense marcar.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A1\*\*\** : afecta els mòduls que estan connectats al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de *bpf\_\**.

*!(A\*\*1\* T1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function continue from S to E".

- **Postcondicions:**

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + #S01 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}(S)$ ).

*#N02 + #S01 0001* : missatge pel canal 2 al Nord: ( $Min = V_{pot}(S)+1$ ).

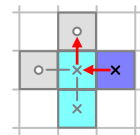


### - Missatges des de l'Est:

<b>Regla:</b>	<i>Spread potential function continue from E to N</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#E05 0001</i> <i>A1***</i> <i>T0,1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>#N05 + 0000 0001</i> <i>#N01 + #E01 0000</i> <i>#N02 + #E01 0001</i>	

### - Regla: Spread potential function continue from E to N

Aquesta regla s'aplica als mòduls als quals arriba (de nou) el missatge de funció potencial, aquest cop per l'Est i han de propagar-lo pel Nord, que forma part d'una altra branca perquè es troba en una bifurcació amb diversos mòduls connectats.



### - Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

### - Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*A1\*\*\** : afecta els mòduls que estan connectats al Nord.

*T0,1, bpf\_\** : afecta els mòduls que tenen el veí del Nord en estat de bpf\_.

### - Postcondicions:

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + #E01 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}(E)$ ).

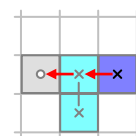
*#N02 + #E01 0001* : missatge pel canal 2 al Nord: ( $Min = V_{pot}(E)+1$ ).

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function continue from E to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#E05 0001</i> <i>A*1**</i> <i>T-1,0, bpf_*</i> <i>!(A1*** T0,1, bpf_*)</i>	
<b>Postcondicions:</b>	
<i>#W05 + 0000 0001</i> <i>#W01 + #E01 0000</i> <i>#W02 + #E01 0001</i>	

- Regla: Spread potential function continue from E to W

*Aquesta regla s'aplica als mòduls als quals arriba (de nou) el missatge de funció potencial, aquest cop per l'Est i han de propagar-lo per l'Oest, que és l'única branca que queda sense marcar.*



- Prioritat: 10000

*La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.*

- Precondicions:

**Sbpf\*\*** : *afecta els mòduls de la fase [bpf]*

**S\*\*\*m\*** : *afecta els mòduls que estat marcats.*

=#E05 0001 : *afecta els mòduls que reben un missatge de "marca" de l'Est.*

**A\*1\*\*** : *afecta els mòduls que estan connectats a l'Oest.*

T-1,0, bpf \* : *afecta els mòduls que tenen el veí de l'Oest en estat de bpf.*

!(A1\*\*\* T0,1, bpf\_\* ) : afecta els mòduls que no apliquen la regla "Spread potential function continue from E to N".

- Postcondiciones:

#W05 + 0000 0001 : *missatge de "marca" a l'Oest.*

#W01 + #E01 0000 : *missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}(E)$ ).*

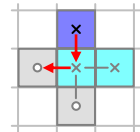
#W02 + #E01 0001 : *missatge pel canal 2 a l'Oest: (Min = Vpot(E)+1).*

### - Missatges des del Nord:

<b>Regla:</b>	<i>Spread potential function continue from N to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#N05 0001</i> <i>A*1**</i> <i>T-1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>#W05 + 0000 0001</i> <i>#W01 + #N01 0000</i> <i>#W02 + #N01 0001</i>	

### - Regla: Spread potential function continue from N to W

Aquesta regla s'aplica als mòduls als quals arriba (de nou) el missatge de funció potencial, aquest cop pel Nord i han de propagar-lo per l'Oest, que forma part d'una altra branca perquè es troba en una bifurcació amb diversos mòduls connectats.



### - Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

### - Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A\*1\*\*\** : afecta els mòduls que estan connectats a l'Oest.

*T-1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *bpf\_\**.

### - Postcondicions:

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + #N01 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}(N)$ ).

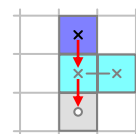
*#W02 + #N01 0001* : missatge pel canal 2 a l'Oest: ( $Min = V_{pot}(N)+1$ ).

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function continue from N to S</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#N05 0001</i> <i>A***1</i> <i>T0,-1, bpf_*</i> <i>!(A*1** T-1,0, bpf_*)</i>	
<b>Postcondicions:</b>	
<i>#S05 + 0000 0001</i> <i>#S01 + #N01 0000</i> <i>#S02 + #N01 0001</i>	

- **Regla:** Spread potential function continue from N to S

Aquesta regla s'aplica als mòduls als quals arriba (de nou) el missatge de funció potencial, aquest cop pel Nord i han de propagar-lo pel Sud, que és l'única branca que queda sense marcar.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*A\*\*\*1* : afecta els mòduls que estan connectats al Sud.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de *bpf\_\**.

*!(A\*1\*\* T-1,0, bpf\_\*)* : afecta els mòduls que no apliquen la regla "Spread potential function continue from N to W".

- **Postcondicions:**

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + #N01 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}(N)$ ).

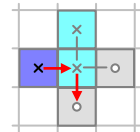
*#S02 + #N01 0001* : missatge pel canal 2 al Sud: ( $Min = V_{pot}(N)+1$ ).

### - Missatges des de l'Oest:

<b>Regla:</b>	<i>Spread potential function continue from W to S</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#W05 0001</i> <i>A***1</i> <i>T0,-1, bpf_*</i>	
<b>Postcondicions:</b>	
<i>#S05 + 0000 0001</i> <i>#S01 + #W01 0000</i> <i>#S02 + #W01 0001</i>	

### - Regla: Spread potential function continue from W to S

Aquesta regla s'aplica als mòduls als quals arriba (de nou) el missatge de funció potencial, aquest cop per l'Oest i han de propagar-lo pel Sud, que forma part d'una altra branca perquè es troba en una bifurcació amb diversos mòduls connectats.



### - Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

### - Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*A\*\*\*1* : afecta els mòduls que estan connectats al Sud.

*T0,-1, bpf\_\** : afecta els mòduls que tenen el veí del Sud en estat de bpf\_.

### - Postcondicions:

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + #W01 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}(W)$ ).

*#S02 + #W01 0001* : missatge pel canal 2 al Sud: ( $Min = V_{pot}(W)+1$ ).

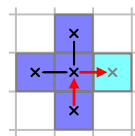


## B.6 Grup: "last"

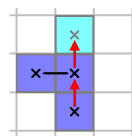
Aquest grup està format per 12 regles subdividides en 4 subgrups de 3 regles. Cada subgrup tracta una de les 4 coordenades per on pot arribar el missatge (N, S, E, W). Cadascun d'aquests subgrups contempla els tres possibles camins on es pot trobar el pare del mòdul.

En aquest grup hem implementat les regles següents:

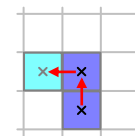
- Spread potential function last



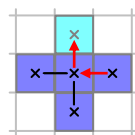
from S to E



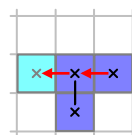
from S to N



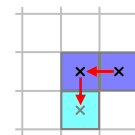
from S to W



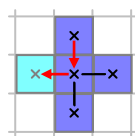
from E to N



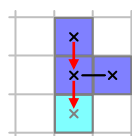
from E to W



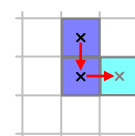
from E to S



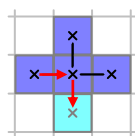
from N to W



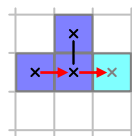
from N to S



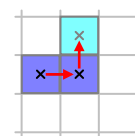
from N to E



from W to S



from W to E



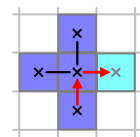
from W to N

- Missatges des del Sud:

<b>Regla:</b>	<i>Spread potential function last from S to E</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> =#S05 0001 = C002 0003	
<b>Postcondicions:</b>	
#E05 + 0000 0001 #E01 + #S01 0000 C004 + #S01 0001 <i>Spfc_*</i>	

- Regla: Spread potential function last from S to E

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial pel Sud i l'han de propagar al seu pare (per l'Est), ja que la resta de branques (si n'hi ha) estan completades.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

=#S05 0001 : afecta els mòduls que reben un missatge de "marca" del Sud.

= C002 0003 : afecta els mòduls que tenen el pare a l'Est.

- Postcondicions:

#E05 + 0000 0001 : missatge de "marca" a l'Est.

#E01 + #S01 0000 : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}(S)$ ).

C004 + #S01 0001 :  $Max = V_{pot}(S) + 1$

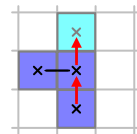
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].



<b>Regla:</b>	<i>Spread potential function last from S to N</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#S05 0001</i> <i>= C002 0001</i> <i>!(A**1* T1,0, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#N05 + 0000 0001</i> <i>#N01 + #S01 0000</i> <i>C004 + #S01 0001</i> <i>Spfc_*</i>	

- Regla: Spread potential function last from S to N

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial pel Sud i l'han de propagar al seu pare (pel Nord), ja que la resta de branques (si n'hi ha) estan completades.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*= C002 0001* : afecta els mòduls que tenen el pare al Nord.

*!(A\*\*1\* T1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from S to E".

- Postcondicions:

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + #S01 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}(S)$ ).

*C004 + #S01 0001* :  $Max = V_{pot}(S) + 1$

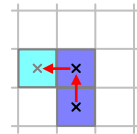
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function last from S to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#S05 0001</i> <i>= C002 0002</i> <i>!(A**1* T1,0, bpf_* )</i> <i>!(A1*** T0,1, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#W05 + 0000 0001</i> <i>#W01 + #S01 0000</i> <i>C004 + #S01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last from S to W

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial pel Sud i l'han de propagar al seu pare (per l'Oest). Només li queda fer el camí de tornada per la seva branca.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*= C002 0002* : afecta els mòduls que tenen el pare a l'Oest.

*!(A\*\*1\* T1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from S to E".

*!(A1\*\*\* T0,1, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from S to N".

- **Postcondicions:**

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + #S01 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}(S)$ ).

*C004 + #S01 0001* :  $Max = V_{pot}(S) + 1$

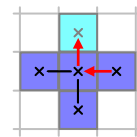
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

### - Missatges des de l'Est:

<b>Regla:</b>	<i>Spread potential function last from E to N</i>
<b>Prioritat:</b>	<i>10000</i>
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#E05 0001</i> <i>= C002 0001</i>	
<b>Postcondicions:</b>	
<i>#N05 + 0000 0001</i> <i>#N01 + #E01 0000</i> <i>C004 + #E01 0001</i> <i>Spfc_*</i>	

- Regla: Spread potential function last from E to N

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial per l'Est i l'han de propagar al seu pare (pel Nord), ja que la resta de branques (si n'hi ha) estan completades.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*= C002 0001* : afecta els mòduls que tenen el pare al Nord.

- Postcondicions:

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + #E01 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}(E)$ ).

*C004 + #E01 0001* :  $Max = V_{pot}(E) + 1$

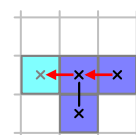
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function last from E to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#E05 0001</i> <i>= C002 0002</i> <i>!(A1*** T0,1, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#W05 + 0000 0001</i> <i>#W01 + #E01 0000</i> <i>C004 + #E01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last from E to W

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial per l'Est i l'han de propagar al seu pare (per l'Oest), ja que la resta de branques (si n'hi ha) estan completades.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*= C002 0002* : afecta els mòduls que tenen el pare a l'Oest.

*!(A1\*\*\* T0,1, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from E to N".

- **Postcondicions:**

*#W05 + 0000 0001* : missatge de "marca" a l'Oest.

*#W01 + #E01 0000* : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}(E)$ ).

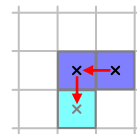
*C004 + #E01 0001* :  $Max = V_{pot}(E) + 1$

*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

<b>Regla:</b>	<i>Spread potential function last from E to S</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#E05 0001</i> <i>= C002 0004</i> <i>!(A1*** T0,1, bpf_* )</i> <i>!(A*1** T-1,0, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#S05 + 0000 0001</i> <i>#S01 + #E01 0000</i> <i>C004 + #E01 0001</i> <i>Spfc_*</i>	

- Regla: Spread potential function last from E to S

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial per l'Est i l'han de propagar al seu pare (pel Sud). Només li queda fer el camí de tornada per la seva branca.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#E05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Est.

*= C002 0004* : afecta els mòduls que tenen el pare al Sud.

*!(A1\*\*\* T0,1, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from E to N".

*!(A\*1\*\* T-1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from E to W".

- Postcondicions:

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + #E01 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}(E)$ ).

*C004 + #E01 0001* :  $Max = V_{pot}(E) + 1$

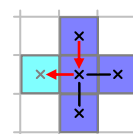
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

- Missatges des del Nord:

<b>Regla:</b>	<i>Spread potential function last from N to W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> =#N05 0001 = C002 0002	
<b>Postcondicions:</b>	
#W05 + 0000 0001 #W01 + #N01 0000 C004 + #N01 0001 <i>Spfc_*</i>	

- Regla: Spread potential function last from N to W

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial pel Nord i l'han de propagar al seu pare (per l'Oest), ja que la resta de branques (si n'hi ha) estan completades.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

=#N05 0001 : afecta els mòduls que reben un missatge de "marca" del Nord.

= C002 0002 : afecta els mòduls que tenen el pare a l'Oest.

- Postcondicions:

#W05 + 0000 0001 : missatge de "marca" a l'Oest.

#W01 + #N01 0000 : missatge pel canal 1 a l'Oest: ( $V_{pot} = V_{pot}(N)$ ).

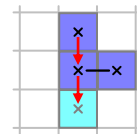
C004 + #N01 0001 :  $Max = V_{pot}(N) + 1$

*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

<b>Regla:</b>	<i>Spread potential function last from N to S</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#N05 0001</i> <i>= C002 0004</i> <i>!(A*1** T-1,0, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#S05 + 0000 0001</i> <i>#S01 + #N01 0000</i> <i>C004 + #N01 0001</i> <i>Spfc_*</i>	

- Regla: Spread potential function last from N to S

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial pel Nord i l'han de propagar al seu pare (pel Sud), ja que la resta de branques (si n'hi ha) estan completades.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*= C002 0004* : afecta els mòduls que tenen el pare al Sud.

*!(A\*1\*\* T-1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from N to W".

- Postcondicions:

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + #N01 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}(N)$ ).

*C004 + #N01 0001* :  $Max = V_{pot}(N) + 1$

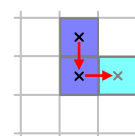
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function last from N to E</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#N05 0001</i> <i>= C002 0003</i> <i>!(A*1** T-1,0, bpf_* )</i> <i>!(A***1 T0,-1, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#E05 + 0000 0001</i> <i>#E01 + #N01 0000</i> <i>C004 + #N01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last from N to E

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial pel Nord i l'han de propagar al seu pare (per l'Est). Només li queda fer el camí de tornada per la seva branca.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*= C002 0003* : afecta els mòduls que tenen el pare a l'Est.

*!(A\*1\*\* T-1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from N to W".

*!(A\*\*\*1 T0,-1, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from N to S".

- **Postcondicions:**

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + #N01 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}(N)$ ).

*C004 + #N01 0001* :  $Max = V_{pot}(N) + 1$

*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

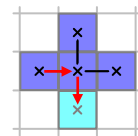


## - Missatges des de l'Oest:

<b>Regla:</b>	<i>Spread potential function last from W to S</i>
<b>Prioritat:</b>	<i>10000</i>
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#W05 0001</i> <i>= C002 0004</i>	
<b>Postcondicions:</b>	
<i>#S05 + 0000 0001</i> <i>#S01 + #W01 0000</i> <i>C004 + #W01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last from W to S

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial per l'Oest i l'han de propagar al seu pare (pel Sud), ja que la resta de branques (si n'hi ha) estan completades.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*= C002 0004* : afecta els mòduls que tenen el pare al Sud.

- **Postcondicions:**

*#S05 + 0000 0001* : missatge de "marca" al Sud.

*#S01 + #W01 0000* : missatge pel canal 1 al Sud: ( $V_{pot} = V_{pot}(W)$ ).

*C004 + #W01 0001* :  $Max = V_{pot}(W) + 1$

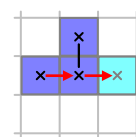
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function last from W to E</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#W05 0001</i> <i>= C002 0003</i> <i>!(A***1 T0,-1, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#E05 + 0000 0001</i> <i>#E01 + #W01 0000</i> <i>C004 + #W01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last from W to E

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial per l'Oest i l'han de propagar al seu pare (per l'Est), ja que la resta de branques (si n'hi ha) estan completades.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*= C002 0003* : afecta els mòduls que tenen el pare a l'Est.

*!(A\*\*\*1 T0,-1, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from W to S".

- **Postcondicions:**

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + #W01 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}(W)$ ).

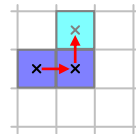
*C004 + #W01 0001* :  $Max = V_{pot}(W) + 1$

*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

<b>Regla:</b>	<i>Spread potential function last from W to N</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>=#W05 0001</i> <i>= C002 0001</i> <i>!(A***1 T0,-1, bpf_* )</i> <i>!(A**1* T1,0, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>#N05 + 0000 0001</i> <i>#N01 + #W01 0000</i> <i>C004 + #W01 0001</i> <i>Spfc_*</i>	

- Regla: Spread potential function last from W to N

Aquesta regla s'aplica als mòduls als quals arriba el missatge de funció potencial per l'Oest i l'han de propagar al seu pare (pel Nord). Només li queda fer el camí de tornada per la seva branca.



- Prioritat: 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- Precondicions:

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

*= C002 0001* : afecta els mòduls que tenen el pare al Nord.

*!(A\*\*\*1 T0,-1, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from W to S".

*!(A\*\*1\* T1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from W to E".

- Postcondicions:

*#N05 + 0000 0001* : missatge de "marca" al Nord.

*#N01 + #W01 0000* : missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}(W)$ ).

*C004 + #W01 0001* :  $Max = V_{pot}(W) + 1$

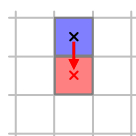
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

### **B.7 Grup: “last father”**

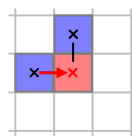
Aquest grup està format per 2 regles que contempen els dos camins pels quals li pot arribar el missatge al màster, pel Nord o per l'Oest.

En aquest grup hem implementat les regles següents:

- Spread potential function last father



*from N*

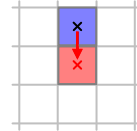


*from W*

<b>Regla:</b>	<i>Spread potential function last father from N</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>S****M</i> <i>=#N05 0001</i> <i>!(A*1** T-1,0, bpf_* )</i>	
<b>Postcondicions:</b>	
<i>C004 + #N01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last father from N

Aquesta regla s'aplica al mòdul màster al qual li arriba el missatge de funció potencial pel Nord i no li queda cap altra branca per completar. Aquesta regla dona per finalitzada la construcció de la funció potencial.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*S\*\*\*\*M* : afecta els mòduls identificats com a màster.

*=#N05 0001* : afecta els mòduls que reben un missatge de "marca" del Nord.

*!(A\*1\*\* T-1,0, bpf\_\* )* : afecta els mòduls que no apliquen la regla "Spread potential function continue from N to W".

- **Postcondicions:**

*C004 + #N01 0001* :  $Max = V_{pot}(N) + 1$

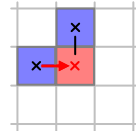
*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function last father from W</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sbpf**</i> <i>S***m*</i> <i>S****M</i> <i>=#W05 0001</i>	
<b>Postcondicions:</b>	
<i>C004 + #W01 0001</i> <i>Spfc_*</i>	

- **Regla:** Spread potential function last father from W

Aquesta regla s'aplica al mòdul màster al qual li arriba el missatge de funció potencial per l'Oest i no li queda cap altra branca per completar. Aquesta regla dóna per finalitzada la construcció de la funció potencial.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini] i [bst] per poder treballar en paral·lel.

- **Precondicions:**

*Sbpf\*\** : afecta els mòduls de la fase [bpf]

*S\*\*\*m\** : afecta els mòduls que estat marcats.

*S\*\*\*\*M* : afecta els mòduls identificats com a màster.

*=#W05 0001* : afecta els mòduls que reben un missatge de "marca" de l'Oest.

- **Postcondicions:**

*C004 + #W01 0001* :  $Max = V_{pot}(W) + 1$

*Spfc\_\** : canvia el seu estat a la nova fase [pfc].

### **B.8 Grup extra: *“first go adapt”* i *“first wait adapt”***

Aquest grup està format per les mateixes 24 + 4 regles que els seus grups anàlegs (*“first go”* i *“first wait”*).

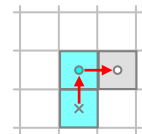
Aquestes regles tenen les mateixes condicions i realitzen la mateixa funció amb els mateixos resultats que les seves regles equivalents. Les úniques diferències en cadascuna de les regles són l'estat inicial i l'estat final del mòdul. L'estat inicial pertany a una fase prèvia. L'estat final, depenent de cada cas, l'actualitza directament a la fase actual en l'estat que li pertoca (marcat, en espera o complet).

En aquest grup només mostrem un exemple com a referència.

## Annex B - Regles de la fase de construcció de la funció potencial

<b>Regla:</b>	<i>Spread potential function first from S to E same branch (ini adapt)</i>
<b>Prioritat:</b>	10000
<b>Precondicions:</b>	
<i>Sini**</i> <i>=#S05 0001</i> <i>A0011</i> <i>T1,0, bpf_*</i>	
<b>Postcondicions:</b>	
<i>C000 + #S01 0001</i> <i>C001 + 0000 0002</i> <i>C002 + 0000 0004</i> <i>C003 + #S02 0000</i> <i>#E05 + 0000 0001</i> <i>#E01 + C000 0000</i> <i>#E02 + C003 0000</i> <i>Sbpfm_</i>	

- **Regla:** Spread potential function first from S to E same branch (ini adapt)  
Aquesta regla s'aplica als mòduls que es troben en la fase [ini] als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar per l'Est, que forma part de la mateixa branca perquè és l'únic altre mòdul al qual estan connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [bst] i [bpf] per poder treballar en paral·lel.

- **Precondicions:**

*Sini\*\** : afecta els mòduls de la fase [ini]

*=#S05 0001* : afecta els mòduls que reben un missatge de "marca" del Sud.

*A0011* : afecta els mòduls que estan connectats només al Sud i a l'Est.

*T1,0, bpf\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *bpf\_*.

- **Postcondicions:**

*C000 + #S01 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0002* : *Signe* = 2

*C002 + 0000 0004* : *Pare* = Sud

*C003 + #S02 0000* : *Min* = *Min*(S)

*#E05 + 0000 0001* : missatge de "marca" a l'Est.

*#E01 + C000 0000* : missatge pel canal 1 a l'Est: ( $V_{pot} = V_{pot}$ ).

*#E02 + C003 0000* : missatge pel canal 2 a l'Est: (*Min* = *Min*).

*Sbpfm\_* : canvia a la fase *bpf* en estat "marcat".



## Annex C

### *Regles de la fase de reconfiguració a la forma canònica*

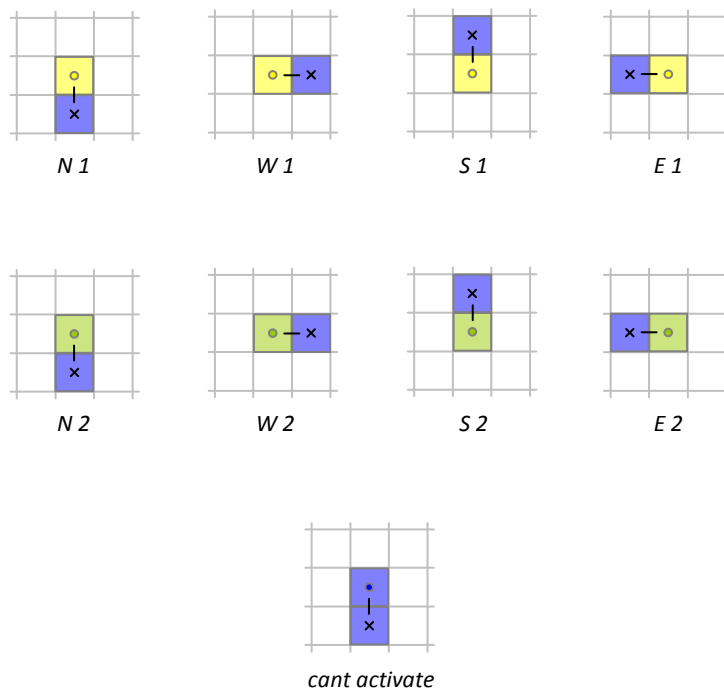
En aquest apartat es detallen les regles de cada grup de la fase de reconfiguració de la forma inicial a la forma canònica. Aquestes regles s'utilitzen per realitzar els moviments dels mòduls que avancen cap el màster i generen la tira [fw] (*Forwards*). L'objectiu, l'estratègia i una visió general d'aquest conjunt de regles es descriuen al Capítol 6.

#### C.1 Grup: "activate"

Aquest grup està format per 9 regles, 8 de les quals estan dividides en 2 subgrups de 4 regles. La regla restant és independent de les altres. El primer subgrup tracta la intencionalitat del mòdul i el segon realitza l'activació. Cadascuna de les regles del subgrup fa referència a cadascuna de les 4 coordenades on es pot trobar la fulla de la branca (N, S, E, W). La regla especial anul·la la intenció del mòdul tornant a l'estat anterior.

En aquest grup hem implementat les regles següents:

- Activate Leaf

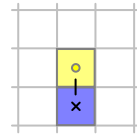


- Regles d'intenció:

Regla:	Activate Leaf N 1
Prioritat:	8022
Precondicions:	<i>Spfc_*</i> <i>!S****M</i> <i>!S****E</i> <i>A0001</i> <i>T0,-1, pfc_*</i>
Postcondicions:	<i>SpfcA*</i>

- Regla: Activate Leaf N 1

Aquesta regla s'aplica a les fulles de l'arbre connectades pel Sud que han de mostrar la seva intenció d'activar-se.



- Prioritat: 8022

La prioritat és més petita que la de les fases [ini] i [bst] perquè primer han d'aplicar les anteriors. La prioritat és més gran que la dels grups de regles de moviment perquè un mòdul s'ha d'activar abans de poder-se moure.

- Precondicions:

*Spfc\_\** : afecta els mòduls de la fase [pfc].

*!S\*\*\*\*M* : afecta els mòduls que no són màster.

*!S\*\*\*\*E* : afecta els mòduls que no formen part de la tirallonga.

*A0001* : afecta els mòduls que estan connectats només al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de *pfc\_\**.

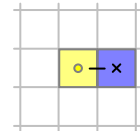
- Postcondicions:

*SpfcA\** : canvia el seu estat a "intenció d'activar-se".

<b>Regla:</b>	Activate Leaf W 1
<b>Prioritat:</b>	8022
<b>Precondicions:</b>	
<p>Spfc_*  !S***M  !S***E  A0010  T1,0, pfc_*</p>	
<b>Postcondicions:</b>	
SpfcA*	

- Regla: Activate Leaf W 1

Aquesta regla s'aplica a les fulles de l'arbre connectades per l'Est que han de mostrar la seva intenció d'activar-se.



- Prioritat: 8022

La prioritat és més petita que la de les fases [ini] i [bst] perquè primer han d'aplicar les anteriors. La prioritat és més gran que la dels grups de regles de moviment perquè un mòdul s'ha d'activar abans de poder-se moure.

- Precondicions:

Spfc\_\* : afecta els mòduls de la fase [pfc].

!S\*\*\*M : afecta els mòduls que no són màster.

!S\*\*\*E : afecta els mòduls que no formen part de la tirallonga.

A0010 : afecta els mòduls que estan connectats només a l'Est.

T1,0, pfc\_\* : afecta els mòduls que tenen el veí de l'Est en estat de pfc\_.

- Postcondicions:

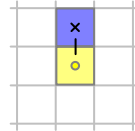
SpfcA\* : canvia el seu estat a "intenció d'activar-se".

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Activate Leaf S 1
<b>Prioritat:</b>	8022
<b>Precondicions:</b>	<i>Spfc_*</i> <i>!S***M</i> <i>!S***E</i> <i>A1000</i> <i>T0,1, pfc_*</i>
<b>Postcondicions:</b>	<i>SpfcA*</i>

### - Regla: Activate Leaf S 1

Aquesta regla s'aplica a les fulles de l'arbre connectades pel Nord que han de mostrar la seva intenció d'activar-se.



### - Prioritat: 8022

La prioritat és més petita que la de les fases [ini] i [bst] perquè primer han d'aplicar les anteriors. La prioritat és més gran que la dels grups de regles de moviment perquè un mòdul s'ha d'activar abans de poder-se moure.

### - Precondicions:

*Spfc\_\** : afecta els mòduls de la fase [pfc].

*!S\*\*\*M* : afecta els mòduls que no són màster.

*!S\*\*\*E* : afecta els mòduls que no formen part de la tirallonga.

*A1000* : afecta els mòduls que estan connectats només al Nord.

*T0,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord en estat de *pfc\_\**.

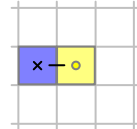
### - Postcondicions:

*SpfcA\** : canvia el seu estat a "intenció d'activar-se".

<b>Regla:</b>	Activate Leaf E 1
<b>Prioritat:</b>	8022
<b>Precondicions:</b>	<p>Spfc_*  !S***M  !S***E  A0100  T-1,0, pfc_*</p>
<b>Postcondicions:</b>	SpfcA*

- Regla: Activate Leaf E 1

Aquesta regla s'aplica a les fulles de l'arbre connectades per l'Oest que han de mostrar la seva intenció d'activar-se.



- Prioritat: 8022

La prioritat és més petita que la de les fases [ini] i [bst] perquè primer han d'aplicar les anteriors. La prioritat és més gran que la dels grups de regles de moviment perquè un mòdul s'ha d'activar abans de poder-se moure.

- Precondicions:

Spfc\_\* : afecta els mòduls de la fase [pfc].

!S\*\*\*M : afecta els mòduls que no són màster.

!S\*\*\*E : afecta els mòduls que no formen part de la tirallonga.

A0100 : afecta els mòduls que estan connectats només a l'Oest.

T-1,0, pfc\_\* : afecta els mòduls que tenen el veí de l'Oest en estat de pfc\_.

- Postcondicions:

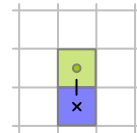
SpfcA\* : canvia el seu estat a "intenció d'activar-se".

**- Regles d'acció:**

<b>Regla:</b>	Activate Leaf N 2
<b>Prioritat:</b>	8021
<b>Precondicions:</b>	
SpfcA* !S****M !S****E A0001 T0,-1, pfc_*	
<b>Postcondicions:</b>	
C000 + 0,-1, C000 0001 C001 + 0000 0001 C003 + 0,-1, C003 0000 C004 + 0,-1, C004 0000 C020 + 0000 0000 C021 + 0000 0000 Sfw__A	

**- Regla:** Activate Leaf N 2

Aquesta regla s'aplica als mòduls connectats pel Sud que tenen la intenció d'activar-se i que res no els impedeix fer-ho.



**- Prioritat:** 8021

La prioritat és més petita que la de les regles d'intenció perquè abans de realitzar l'acció els mòduls han de comunicar la seva intenció.

**- Precondicions:**

SpfcA\* : afecta els mòduls de la fase [pfc] amb intenció d'activar-se.

!S\*\*\*\*M : afecta els mòduls que no són màster.

!S\*\*\*\*E : afecta els mòduls que no formen part de la tirallonga.

A0001 : afecta els mòduls que estan connectats només al Sud.

T0,-1, pfc\_\* : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

**- Postcondicions:**

C000 + 0,-1, C000 0001 :  $Vpot = Vpot(S) + 1$

C001 + 0000 0001 :  $Signe = 1$

C003 + 0,-1, C003 0000 :  $Min = Min(S)$

C004 + 0,-1, C004 0000 :  $Max = Max(S)$

C020 + 0000 0000 :  $Coordenada\ x = 0$

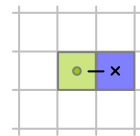
C021 + 0000 0000 :  $Coordenada\ y = 0$

Sfw\_\_A : canvia el seu estat a la nova fase [fw] i a mode "actiu".

<b>Regla:</b>	Activate Leaf W 2
<b>Prioritat:</b>	8021
<b>Precondicions:</b>	
<p>SpfcA*</p> <p>!S****M</p> <p>!S****E</p> <p>A0010</p> <p>T1,0, pfc_*</p>	
<b>Postcondicions:</b>	
<p>C000 + 1,0, C000 0001</p> <p>C001 + 0000 0001</p> <p>C003 + 1,0, C003 0000</p> <p>C004 + 1,0, C004 0000</p> <p>C020 + 0000 0000</p> <p>C021 + 0000 0000</p> <p>Sfw__A</p>	

- Regla: Activate Leaf W 2

Aquesta regla s'aplica als mòduls connectats per l'Est que tenen la intenció d'activar-se i que res no els impedeix fer-ho.



- Prioritat: 8021

La prioritat és més petita que la de les regles d'intenció perquè abans de realitzar l'acció els mòduls han de comunicar la seva intenció.

- Precondicions:

SpfcA\* : afecta els mòduls de la fase [pfc] amb intenció d'activar-se.

!S\*\*\*\*M : afecta els mòduls que no són màster.

!S\*\*\*\*E : afecta els mòduls que no formen part de la tirallonga.

A0010 : afecta els mòduls que estan connectats només a l'Est.

T1,0, pfc\_\* : afecta els mòduls que tenen el veí de l'Est en estat de pfc\_.

- Postcondicions:

C000 + 1,0, C000 0001 :  $V_{pot} = V_{pot}(E) + 1$

C001 + 0000 0001 :  $Signe = 1$

C003 + 1,0, C003 0000 :  $Min = Min(E)$

C004 + 1,0, C004 0000 :  $Max = Max(E)$

C020 + 0000 0000 :  $Coordenada\ x = 0$

C021 + 0000 0000 :  $Coordenada\ y = 0$

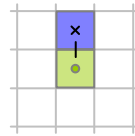
Sfw\_\_A : canvia el seu estat a la nova fase [fw] i a mode "actiu".

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Activate Leaf S 2
<b>Prioritat:</b>	8021
<b>Precondicions:</b>	
SpfcA* !S***M !S***E A1000 T0,1, pfc_*	
<b>Postcondicions:</b>	
C000 + 0,1, C000 0001 C001 + 0000 0001 C003 + 0,1, C003 0000 C004 + 0,1, C004 0000 C020 + 0000 0000 C021 + 0000 0000 Sfw__A	

### - Regla: Activate Leaf S 2

Aquesta regla s'aplica als mòduls connectats pel Nord que tenen la intenció d'activar-se i que res no els impedeix fer-ho.



### - Prioritat: 8021

La prioritat és més petita que la de les regles d'intenció perquè abans de realitzar l'acció els mòduls han de comunicar la seva intenció.

### - Precondicions:

SpfcA\* : afecta els mòduls de la fase [pfc] amb intenció d'activar-se.

!S\*\*\*M : afecta els mòduls que no són màster.

!S\*\*\*E : afecta els mòduls que no formen part de la tirallonga.

A1000 : afecta els mòduls que estan connectats només al Nord.

T0,1, pfc\_\* : afecta els mòduls que tenen el veí del Nord en estat de pfc\_.

### - Postcondicions:

C000 + 0,1, C000 0001 :  $V_{pot} = V_{pot}(N) + 1$

C001 + 0000 0001 :  $Signe = 1$

C003 + 0,1, C003 0000 :  $Min = Min(N)$

C004 + 0,1, C004 0000 :  $Max = Max(N)$

C020 + 0000 0000 :  $Coordenada\ x = 0$

C021 + 0000 0000 :  $Coordenada\ y = 0$

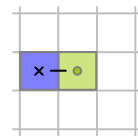
Sfw\_\_A : canvia el seu estat a la nova fase [fw] i a mode "actiu".



<b>Regla:</b>	Activate Leaf E 2
<b>Prioritat:</b>	8021
<b>Precondicions:</b>	
<p><i>SpfcA*</i>  <i>!S****M</i>  <i>!S****E</i>  <i>A0100</i>  <i>T-1,0, pfc_*</i></p>	
<b>Postcondicions:</b>	
<p><i>C000 + -1,0, C000 0001</i>  <i>C001 + 0000 0001</i>  <i>C003 + -1,0, C003 0000</i>  <i>C004 + -1,0, C004 0000</i>  <i>C020 + 0000 0000</i>  <i>C021 + 0000 0000</i>  <i>Sfw__A</i></p>	

- Regla: Activate Leaf E 2

Aquesta regla s'aplica als mòduls connectats per l'Oest que tenen la intenció d'activar-se i que res no els impedeix fer-ho.



- Prioritat: 8021

La prioritat és més petita que la de les regles d'intenció perquè abans de realitzar l'acció els mòduls han de comunicar la seva intenció.

- Precondicions:

*SpfcA\** : afecta els mòduls de la fase [pfc] amb intenció d'activar-se.

*!S\*\*\*\*M* : afecta els mòduls que no són màster.

*!S\*\*\*\*E* : afecta els mòduls que no formen part de la tirallonga.

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*T-1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *pfc\_*.

- Postcondicions:

*C000 + -1,0, C000 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + -1,0, C003 0000* :  $Min = Min(W)$

*C004 + -1,0, C004 0000* :  $Max = Max(W)$

*C020 + 0000 0000* :  $Coordenada\ x = 0$

*C021 + 0000 0000* :  $Coordenada\ y = 0$

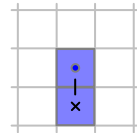
*Sfw\_\_A* : canvia el seu estat a la nova fase [fw] i a mode "actiu".

**- Regla d'anul·lació:**

<b>Regla:</b>	<i>Activate Leaf cant activate</i>
<b>Prioritat:</b>	<i>8020</i>
<b>Precondicions:</b>	
	<i>SpfcA*</i>
<b>Postcondicions:</b>	
	<i>Spfc_*</i>

**- Regla:** Activate Leaf cant activate

*Aquesta regla s'aplica als mòduls que tenen la intenció d'activar-se però que alguna raó els impedeix fer-ho, com ara que algun mòdul els impedeixi activar.*



**- Prioritat:** 8020

*La prioritat és més petita que la de les regles d'acció perquè en cas de no poder fer efectiva l'activació, el mòdul torna al seu estat anterior.*

**- Precondicions:**

*SpfcA\* : afecta els mòduls de la fase [pfc] amb intenció d'activar-se.*

**- Postcondicions:**

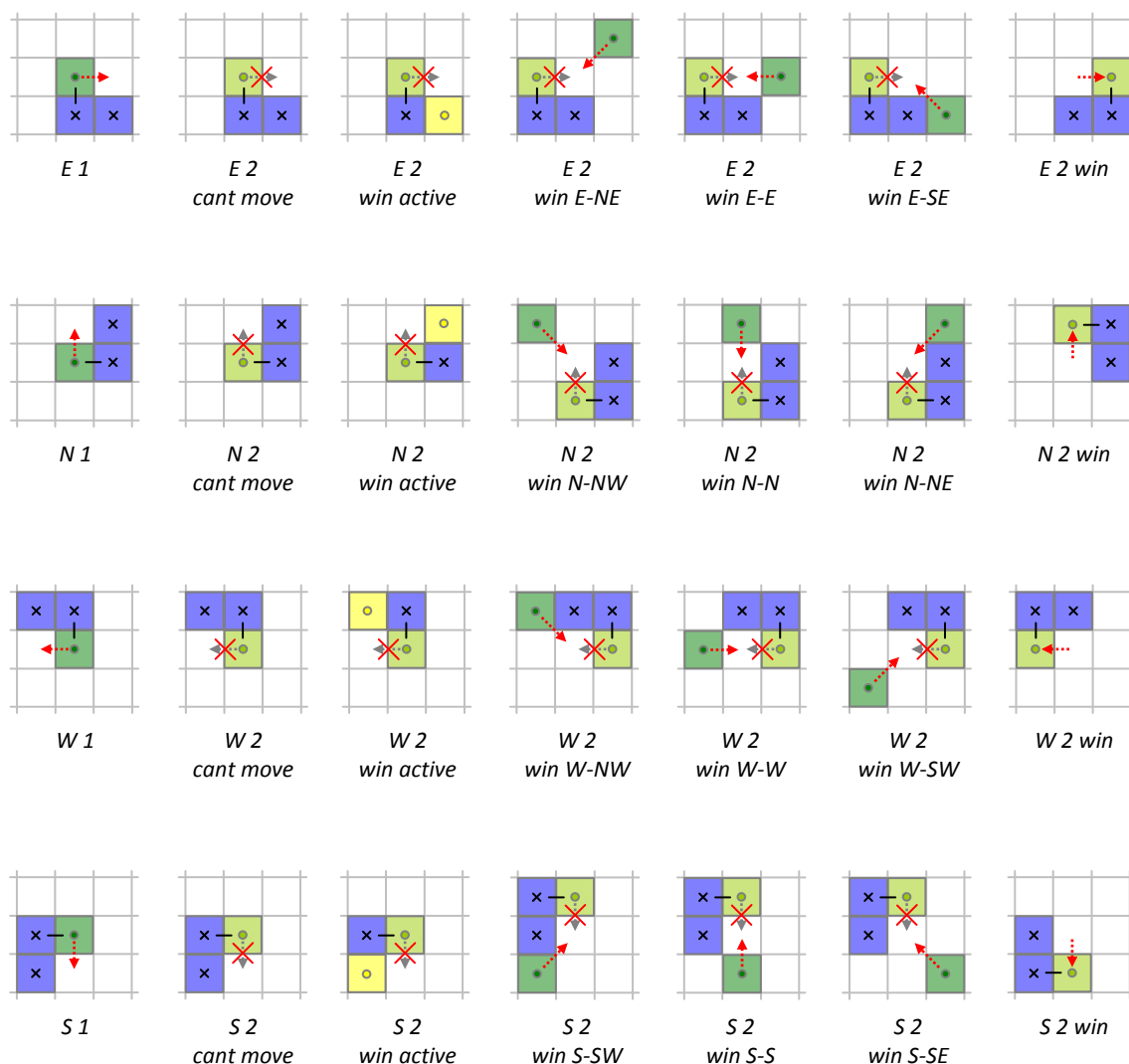
*Spfc\_\* : torna el seu estat a la fase [pfc].*

## C.2 Grup: "slide"

Aquest grup està format per 28 regles subdividides en 4 subgrups de 7 regles. Cadascun dels subgrups fa referència a cadascuna de les 4 coordenades cap on el mòdul vol realitzar el moviment (N, S, E, W). Cadascun d'aquests subgrups té 1 regla d'intencionalitat, 5 regles de verificació i 1 regla d'acció.

En aquest grup hem implementat les regles següents:

- Slide

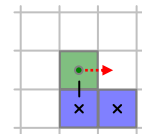


- **Moviment cap a l'Est (intenció):**

<b>Regla:</b>	Slide E 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>E1,0</i> <i>A0001</i> <i>F1,-1</i> <i>T1,-1, pfc_*</i> <i>!T1,-1, ****A</i> <i>V1,-1, C000 C000</i> <i>!( !(E0,1 ) !(F0,1 !( !(T0,1, pfc_* ) !(T0,1, ****A ) ) ) ) )</i>	
<b>Postcondicions:</b>	
<i>SfwE_A</i>	

- **Regla:** Slide E 1

Aquesta regla s'aplica als mòduls actius connectats al Sud que tenen possibilitats de moure's a l'Est i mostren aquesta intenció de moviment.



- **Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Slide és potencialment més petita.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*E1,0* : afecta els mòduls que tenen un espai buit a l'Est.

*A0001* : afecta els mòduls que estan connectats només al Sud.

*F1,-1* : afecta els mòduls que tenen un veí al Sud-Est.

*T1,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud-Est en estat de *pfc\_\**.

*!T1,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud-Est en mode estàtic.

*V1,-1, C000 C000* : afecta els mòduls que tenen el veí del Sud-Est amb valor potencial més petit.

*!( !(E0,1 ) !(F0,1 !( !(T0,1, pfc\_\* ) !(T0,1, \*\*\*\*A ) ) ) )* : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

- **Postcondicions:**

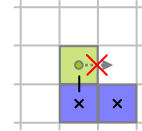
*SfwE\_A* : canvia el seu estat a "intenció de moviment a l'Est".

**- Moviment cap a l'Est (verificació):**

<b>Regla:</b>	Slide E 2 cant move
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	
SfwE_A !(E1,0 A0001 F1,-1 T1,-1, pfc_* !T1,-1, ****A V1,-1, C000 C000 )	
<b>Postcondicions:</b>	
Sfw__A	

**- Regla:** Slide E 2 cant move

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Est, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.

**- Prioritat:** 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

**- Precondicions:**

SfwE\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Est.

!(E1,0 A0001 F1,-1 T1,-1, pfc\_\* !T1,-1, \*\*\*\*A V1,-1, C000 C000 ) : afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.

**- Postcondicions:**

Sfw\_\_A : torna el seu estat a mode d'espera.

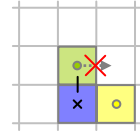
## Annex C - Regles de la fase de reconfiguració a la forma canònica

---

<b>Regla:</b>	Slide E 2 win active
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwE_A</i> <i>F1,-1</i> <i>T1,-1, pfcA_</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Slide E 2 win active

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Est que ja no poden fer-ho perquè el mòdul on havien d'anar és vol activar.



### - Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwE\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Est.

*F1,-1* : afecta els mòduls que tenen un veí al Sud-Est.

*T1,-1, pfcA\_* : afecta els mòduls que tenen el veí del Sud-Est amb intenció d'activar-se.

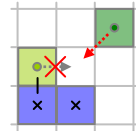
### - Postcondicions:

*Sfw\_\_A* : torna el seu estat a mode d'espera.

<b>Regla:</b>	Slide E 2 win E-NE
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwE_A</i> <i>F2,1</i> <i>T2,1, fwSWA</i> <i>V2,1, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

- Regla: Slide E 2 win E-NE

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwE\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Est.

*F2,1* : afecta els mòduls que tenen un veí a l'Est Nord-Est.

*T2,1, fwSWA* : afecta els mòduls que tenen un veí a l'Est Nord-Est amb intenció de moure's a la mateixa posició.

*V2,1, C000 C000* : afecta els mòduls que tenen un veí a l'Est Nord-Est amb un valor potencial més petit.

- Postcondicions:

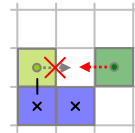
*Sfw\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Slide E 2 win E-E
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwE_A</i> <i>F2,0</i> <i>T2,0, fwW_A</i> <i>V2,0, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

### - Regla: Slide E 2 win E-E

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



### - Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwE\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Est.

*F2,0* : afecta els mòduls que tenen un veí a l'Est-Est.

*T2,0, fwW\_A* : afecta els mòduls que tenen un veí a l'Est-Est amb intenció de moure's a la mateixa posició.

*V2,0, C000 C000* : afecta els mòduls que tenen un veí a l'Est-Est amb un valor potencial més petit.

### - Postcondicions:

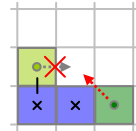
*Sfw\_A* : torna el seu estat a mode d'espera.



<b>Regla:</b>	Slide E 2 win E-SE
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwE_A</i> <i>F2,-1</i> <i>T2,-1, fwNWA</i> <i>V2,-1, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

- Regla: Slide E 2 win E-SE

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwE\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Est.

*F2,-1* : afecta els mòduls que tenen un veí a l'Est Sud-Est.

*T2,-1, fwNWA* : afecta els mòduls que tenen un veí a l'Est Sud-Est amb intenció de moure's a la mateixa posició.

*V2,-1, C000 C000* : afecta els mòduls que tenen un veí a l'Est Sud-Est amb un valor potencial més petit.

- Postcondicions:

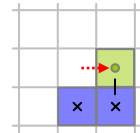
*Sfw\_A* : torna el seu estat a mode d'espera.

**- Moviment cap a l'Est (acció):**

<b>Regla:</b>	Slide E 2 win
<b>Prioritat:</b>	8001
<b>Precondicions:</b>	
SfwE_A	
<b>Postcondicions:</b>	
A000* P1,0 C000 + 0,-1, C000 0001 C001 + 0000 0001 C003 + 0,-1, C003 0000 C004 + 0,-1, C004 0000 C020 + C020 0001 C021 + C021 0000 Sfw__A	

**- Regla:** Slide E 2 win

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Est i que res no els impedeix fer-ho.



**- Prioritat:** 8001

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.

**- Precondicions:**

SfwE\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Est.

**- Postcondicions:**

A000\* : es desconnecta del Sud i es torna a connectar després de moure's.

P1,0 : es mou una posició en direcció Est.

C000 + 0,-1, C000 0001 :  $V_{pot} = V_{pot}(S) + 1$

C001 + 0000 0001 :  $Signe = 1$

C003 + 0,-1, C003 0000 :  $Min = Min(S)$

C004 + 0,-1, C004 0000 :  $Max = Max(S)$

C020 + C020 0001 :  $Coordenada\ x = x + 1$

C021 + C021 0000 :  $Coordenada\ y = y + 0$

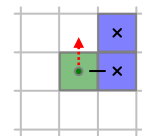
Sfw\_\_A : torna el seu estat a mode d'espera.

**- Moviment cap al Nord (intenció):**

<b>Regla:</b>	Slide N 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>E0,1</i> <i>A0010</i> <i>F1,1</i> <i>T1,1, pfc_*</i> <i>!T1,1, ****A</i> <i>V1,1, C000 C000</i> <i>!( !(E-1,0 ) !(F-1,0 !( !(T-1,0, pfc_ * ) !(T-1,0, ****A ) ) ) ) )</i>	
<b>Postcondicions:</b>	
<i>SfwN_A</i>	

**- Regla:** Slide N 1

Aquesta regla s'aplica als mòduls actius connectats l'Est que tenen possibilitats de moure's al Nord i mostren aquesta intenció de moviment.

**- Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Slide és potencialment més petita.

**- Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*E0,1* : afecta els mòduls que tenen un espai buit al Nord.

*A0010* : afecta els mòduls que estan connectats només a l'Est.

*F1,1* : afecta els mòduls que tenen un veí al Nord-Est.

*T1,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord-Est en estat de *pfc\_\**.

*!T1,1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Nord-Est en mode estàtic.

*V1,1, C000 C000* : afecta els mòduls que tenen el veí del Nord-Est amb valor potencial més petit.

*!( !(E-1,0 ) !(F-1,0 !( !(T-1,0, pfc\_ \* ) !(T-1,0, \*\*\*\*A ) ) ) ) )* : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

**- Postcondicions:**

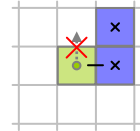
*SfwN\_A* : canvia el seu estat a "intenció de moviment al Nord".

**- Moviment cap al Nord (verificació):**

<b>Regla:</b>	Slide N 2 cant move
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	$SfwN\_A$ $!(E0,1 A0010 F1,1 T1,1, pfc\_ * !T1,1, ****A V1,1, C000 C000 )$
<b>Postcondicions:</b>	$Sfw\_A$

**- Regla:** Slide N 2 cant move

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.



**- Prioritat:** 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

**- Precondicions:**

$SfwN\_A$  : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord.

$!(E0,1 A0010 F1,1 T1,1, pfc\_ * !T1,1, ****A V1,1, C000 C000 )$  : afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.

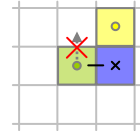
**- Postcondicions:**

$Sfw\_A$  : torna el seu estat a mode d'espera.

<b>Regla:</b>	<i>Slide N 2 win active</i>
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwN_A</i> <i>F1,1</i> <i>T1,1, pfcA_</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Slide N 2 win active

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord que ja no poden fer-ho perquè el mòdul on havien d'anar és vol activar.*



- Prioritat: 8002

*La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

- Precondicions:

*SfwN\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord.*

*F1,1 : afecta els mòduls que tenen un veí al Nord-Est.*

*T1,1, pfcA\_ : afecta els mòduls que tenen el veí del Nord-Est amb intenció d'activar-se.*

- Postcondicions:

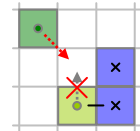
*Sfw\_\_A : torna el seu estat a mode d'espera.*

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Slide N 2 win N-NW</i>
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwN_A</i> <i>F-1,2</i> <i>T-1,2, fwSEA</i> <i>V-1,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Slide N 2 win N-NW

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8002

*La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwN\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord.*

*F-1,2 : afecta els mòduls que tenen un veí al Nord Nord-Oest.*

*T-1,2, fwSEA : afecta els mòduls que tenen un veí al Nord Nord-Oest amb intenció de moure's a la mateixa posició.*

*V-1,2, C000 C000 : afecta els mòduls que tenen un veí al Nord Nord-Oest amb un valor potencial més petit.*

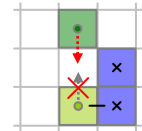
### - Postcondicions:

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Slide N 2 win N-N
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<p>SfwN_A</p> <p>F0,2</p> <p>T0,2, fwS_A</p> <p>V0,2, C000 C000</p>
<b>Postcondicions:</b>	Sfw_A

- Regla: Slide N 2 win N-N

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

SfwN\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord.

F0,2 : afecta els mòduls que tenen un veí al Nord-Nord.

T0,2, fwS\_A : afecta els mòduls que tenen un veí al Nord-Nord amb intenció de moure's a la mateixa posició.

V0,2, C000 C000 : afecta els mòduls que tenen un veí al Nord-Nord amb un valor potencial més petit.

- Postcondicions:

Sfw\_A : torna el seu estat a mode d'espera.



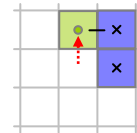


**- Moviment cap al Nord (acció):**

<b>Regla:</b>	<i>Slide N 2 win</i>
<b>Prioritat:</b>	<i>8001</i>
<b>Precondicions:</b>	
<i>SfwN_A</i>	
<b>Postcondicions:</b>	
<i>A00*0</i> <i>P0,1</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i> <i>C020 + C020 0000</i> <i>C021 + C021 0001</i> <i>Sfw__A</i>	

**- Regla:** Slide N 2 win

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord i que res no els impedeix fer-ho.

**- Prioritat:** 8001

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.

**- Precondicions:**

*SfwN\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord.

**- Postcondicions:**

*A00\*0* : es desconnecta de l'Est i es torna a connectar després de moure's.

*P0,1* : es mou una posició en direcció Nord.

*C000 + 1,0, C000 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 1,0, C003 0000* :  $Min = Min(E)$

*C004 + 1,0, C004 0000* :  $Max = Max(E)$

*C020 + C020 0000* :  $Coordenada\ x = x + 0$

*C021 + C021 0001* :  $Coordenada\ y = y + 1$

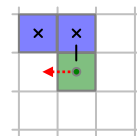
*Sfw\_\_A* : torna el seu estat a mode d'espera.

- **Moviment cap a l'Oest (intenció):**

<b>Regla:</b>	Slide W 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>E-1,0</i> <i>A1000</i> <i>F-1,1</i> <i>T-1,1, pfc_*</i> <i>!T-1,1, ****A</i> <i>V-1,1, C000 C000</i> <i>!( !(E0,-1 ) !(F0,-1 !( !(T0,-1, pfc_* ) !(T0,-1, ****A ) ) ) ) )</i>	
<b>Postcondicions:</b>	
<i>SfwW_A</i>	

- **Regla:** Slide W 1

Aquesta regla s'aplica als mòduls actius connectats al Nord que tenen possibilitats de moure's a l'Oest i mostren aquesta intenció de moviment.



- **Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Slide és potencialment més petita.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*E-1,0* : afecta els mòduls que tenen un espai buit a l'Oest.

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F-1,1* : afecta els mòduls que tenen un veí al Nord-Oest.

*T-1,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord-Oest en estat de *pfc\_\**.

*!T-1,1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Nord-Oest en mode estàtic.

*V-1,1, C000 C000* : afecta els mòduls que tenen el veí del Nord-Oest amb valor potencial més petit.

*!( !(E0,-1 ) !(F0,-1 !( !(T0,-1, pfc\_\* ) !(T0,-1, \*\*\*\*A ) ) ) ) )* : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

- **Postcondicions:**

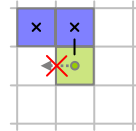
*SfwW\_A* : canvia el seu estat a "intenció de moviment a l'Oest".

### - Moviment cap a l'Oest (verificació):

<b>Regla:</b>	Slide W 2 cant move
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwW_A</i> <i>!(E-1,0 A1000 F-1,1 T-1,1, pfc_* !T-1,1, ****A V-1,1, C000 C000 )</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

#### - Regla: Slide W 2 cant move

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Oest, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.



#### - Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

#### - Precondicions:

*SfwW\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Oest.

*!(E-1,0 A1000 F-1,1 T-1,1, pfc\_\* !T-1,1, \*\*\*\*A V-1,1, C000 C000 )* : afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.

#### - Postcondicions:

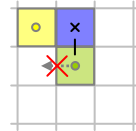
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Slide W 2 win active</i>
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwW_A</i> <i>F-1,1</i> <i>T-1,1, pfcA_</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Slide W 2 win active

*Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Oest que ja no poden fer-ho perquè el mòdul on havien d'anar és vol activar.*



### - Prioritat: 8002

*La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwW\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Oest.*

*F-1,1 : afecta els mòduls que tenen un veí al Nord-Oest.*

*T-1,1, pfcA\_ : afecta els mòduls que tenen el veí del Nord-Oest amb intenció d'activar-se.*

### - Postcondicions:

*Sfw\_\_A : torna el seu estat a mode d'espera.*

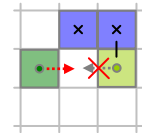
Sfw\_\_A : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Slide W 2 win W-W</i>
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwW_A</i> <i>F-2,0</i> <i>T-2,0, fwE_A</i> <i>V-2,0, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

### - Regla: Slide W 2 win W-W

*Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8002

*La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwW\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Oest.*

*F-2,0 : afecta els mòduls que tenen un veí a l'Oest-Oest.*

*T-2,0, fwE\_A : afecta els mòduls que tenen un veí a l'Oest-Oest amb intenció de moure's a la mateixa posició.*

*V-2,0, C000 C000 : afecta els mòduls que tenen un veí a l'Oest-Oest amb un valor potencial més petit.*

### - Postcondicions:

*Sfw\_A : torna el seu estat a mode d'espera.*

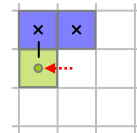
Sfw\_\_A : torna el seu estat a mode d'espera.

**- Moviment cap a l'Oest (acció):**

<b>Regla:</b>	Slide W 2 win
<b>Prioritat:</b>	8001
<b>Precondicions:</b>	<i>SfwW_A</i>
<b>Postcondicions:</b>	<i>A*000</i> <i>P-1,0</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i> <i>C020 - C020 0001</i> <i>C021 + C021 0000</i> <i>Sfw__A</i>

**- Regla:** Slide W 2 win

Aquesta regla s'aplica als mòduls amb intenció de moure's a l'Oest i que res no els impedeix fer-ho.



**- Prioritat:** 8001

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.

**- Precondicions:**

*SfwW\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's a l'Oest.

**- Postcondicions:**

*A\*000* : es desconnecta del Nord i es torna a connectar després de moure's.

*P-1,0* : es mou una posició en direcció Oest.

*C000 + 0,1, C000 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,1, C003 0000* :  $Min = Min(N)$

*C004 + 0,1, C004 0000* :  $Max = Max(N)$

*C020 - C020 0001* :  $Coordenada\ x = x - 1$

*C021 + C021 0000* :  $Coordenada\ y = y + 0$

*Sfw\_\_A* : torna el seu estat a mode d'espera.

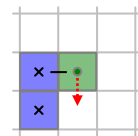


**- Moviment cap al Sud (intenció):**

<b>Regla:</b>	Slide S 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>E0,-1</i> <i>A0100</i> <i>F-1,-1</i> <i>T-1,-1, pfc_*</i> <i>!T-1,-1, ****A</i> <i>V-1,-1, C000 C000</i> <i>!( !(E1,0 ) !(F1,0 !( !(T1,0, pfc_* ) !(T1,0, ****A ) ) ) ) )</i>	
<b>Postcondicions:</b>	
<i>SfwS_A</i>	

**- Regla: Slide S 1**

Aquesta regla s'aplica als mòduls actius connectats l'Oest que tenen possibilitats de moure's al Sud i mostren aquesta intenció de moviment.

**- Prioritat: 8003**

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Slide és potencialment més petita.

**- Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*E0,-1* : afecta els mòduls que tenen un espai buit al Sud.

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*F-1,-1* : afecta els mòduls que tenen un veí al Sud-Oest.

*T-1,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud-Oest en estat de *pfc\_\**.

*!T-1,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud-Oest en mode estàtic.

*V-1,-1, C000 C000* : afecta els mòduls que tenen el veí del Sud-Oest amb valor potencial més petit.

*!( !(E1,0 ) !(F1,0 !( !(T1,0, pfc\_\* ) !(T1,0, \*\*\*\*A ) ) ) )* : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

**- Postcondicions:**

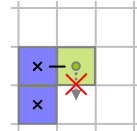
*SfwS\_A* : canvia el seu estat a "intenció de moviment al Sud".

**- Moviment cap al Sud (verificació):**

<b>Regla:</b>	Slide S 2 cant move
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	$SfwS\_A$ $!(E0,-1 A0100 F-1,-1 T-1,-1, pfc\_ * !T-1,-1, ****A V-1,-1, C000 C000 )$
<b>Postcondicions:</b>	$Sfw\_A$

**- Regla:** Slide S 2 cant move

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.



**- Prioritat:** 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

**- Precondicions:**

$SfwS\_A$  : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud.

$!(E0,-1 A0100 F-1,-1 T-1,-1, pfc\_ * !T-1,-1, ****A V-1,-1, C000 C000 )$  : afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.

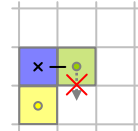
**- Postcondicions:**

$Sfw\_A$  : torna el seu estat a mode d'espera.

<b>Regla:</b>	Slide S 2 win active
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<p>SfwS_A</p> <p>F-1,-1</p> <p>T-1,-1, pfcA_</p>
<b>Postcondicions:</b>	Sfw__A

- Regla: Slide S 2 win active

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud que ja no poden fer-ho perquè el mòdul on havien d'anar és vol activar.



- Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

SfwS\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud.

F-1,-1 : afecta els mòduls que tenen un veí al Sud-Oest.

T-1,-1, pfcA\_ : afecta els mòduls que tenen el veí del Sud-Oest amb intenció d'activar-se.

- Postcondicions:

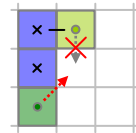
Sfw\_\_A : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Slide S 2 win S-SW
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwS_A</i> <i>F-1,-2</i> <i>T-1,-2, fwNEA</i> <i>V-1,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

### - Regla: Slide S 2 win S-SW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



### - Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwS\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud.

*F-1,-2* : afecta els mòduls que tenen un veí al Sud Sud-Oest.

*T-1,-2, fwNEA* : afecta els mòduls que tenen un veí al Sud Sud-Oest amb intenció de moure's a la mateixa posició.

*V-1,-2, C000 C000* : afecta els mòduls que tenen un veí al Sud Sud-Oest amb un valor potencial més petit.

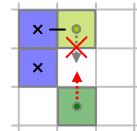
### - Postcondicions:

*Sfw\_A* : torna el seu estat a mode d'espera.

<b>Regla:</b>	Slide S 2 win S-S
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwS_A</i> <i>F0,-2</i> <i>T0,-2, fwE_A</i> <i>V0,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

- Regla: Slide S 2 win S-S

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8002

La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwS\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud.

*F0,-2* : afecta els mòduls que tenen un veí al Sud-Sud.

*T0,-2, fwE\_A* : afecta els mòduls que tenen un veí al Sud-Sud amb intenció de moure's a la mateixa posició.

*V0,-2, C000 C000* : afecta els mòduls que tenen un veí al Sud-Sud amb un valor potencial més petit.

- Postcondicions:

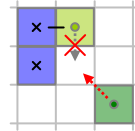
*Sfw\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Slide S 2 win S-SE</i>
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	<i>SfwS_A</i> <i>F1,-2</i> <i>T1,-2, fwNWA</i> <i>V1,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw_A</i>

### - Regla: Slide S 2 win S-SE

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8002

*La prioritat és més petita que la de les regles d'intenció perquè abans de verificar si pot o no realitzar l'acció els mòduls han de comunicar prèviament la seva intenció. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwS\_A : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud.*

*F1,-2 : afecta els mòduls que tenen un veí al Sud Sud-Est.*

*T1,-2, fwNWA : afecta els mòduls que tenen un veí al Sud Sud-Est amb intenció de moure's a la mateixa posició.*

*V1,-2, C000 C000 : afecta els mòduls que tenen un veí al Sud Sud-Est amb un valor potencial més petit.*

### - Postcondicions:

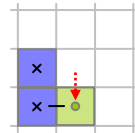
*Sfw\_A : torna el seu estat a mode d'espera.*

**- Moviment cap al Sud (acció):**

<b>Regla:</b>	<i>Slide S 2 win</i>
<b>Prioritat:</b>	<i>8001</i>
<b>Precondicions:</b>	
<i>SfwS_A</i>	
<b>Postcondicions:</b>	
<i>A0*00</i> <i>P0,-1</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i> <i>C020 + C020 0000</i> <i>C021 - C021 0001</i> <i>Sfw__A</i>	

**- Regla:** Slide S 2 win

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud i que res no els impedeix fer-ho.

**- Prioritat:** 8001

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.

**- Precondicions:**

*SfwN\_A* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud.

**- Postcondicions:**

*A0\*00* : es desconnecta de l'Oest i es torna a connectar després de moure's.

*P0,-1* : es mou una posició en direcció Sud.

*C000 + -1,0, C000 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + -1,0, C003 0000* :  $Min = Min(W)$

*C004 + -1,0, C004 0000* :  $Max = Max(W)$

*C020 + C020 0000* :  $Coordenada\ x = x + 0$

*C021 - C021 0001* :  $Coordenada\ y = y - 1$

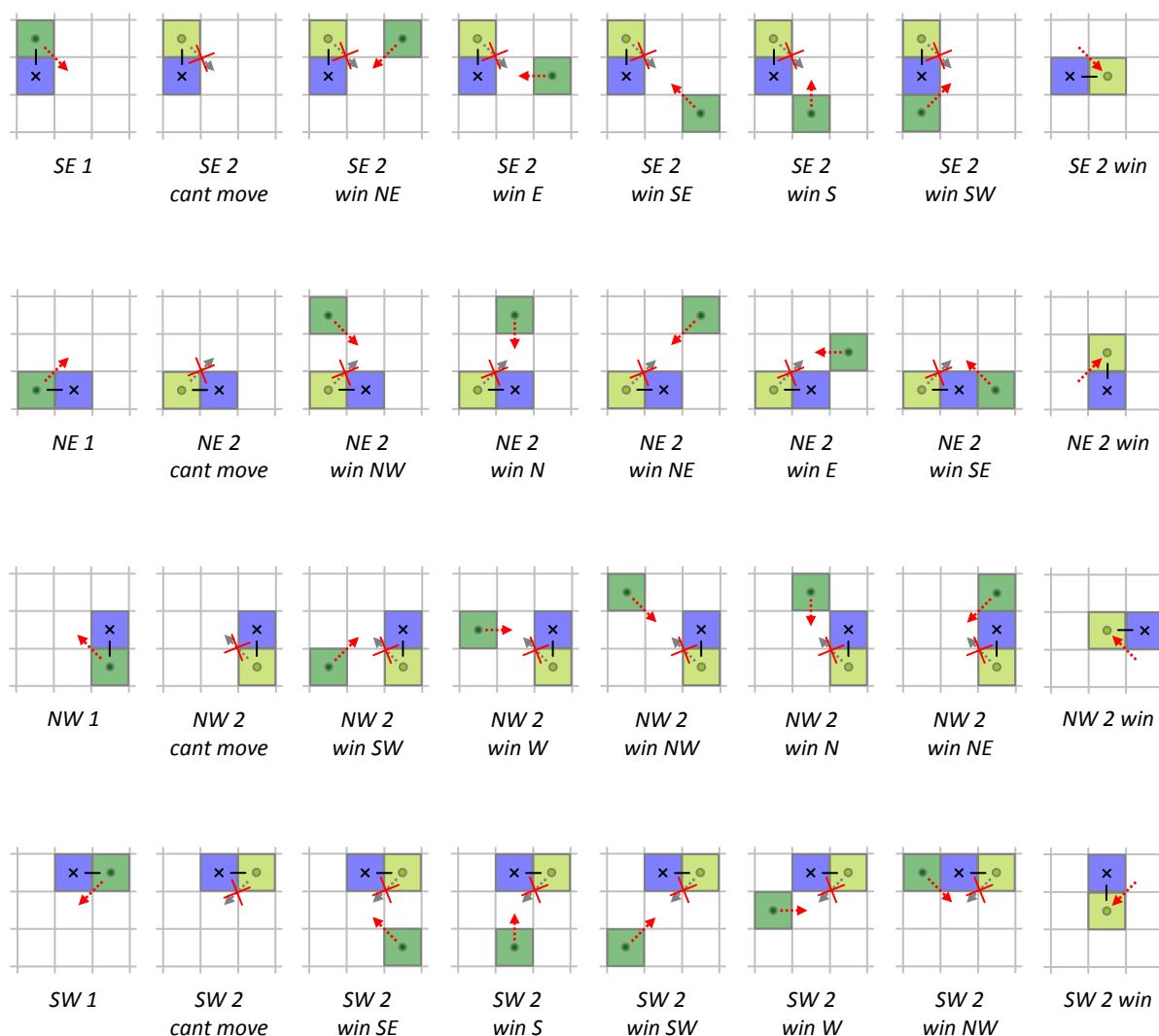
*Sfw\_\_A* : torna el seu estat a mode d'espera.

### C.3 Grup: “convex transition”

Aquest grup està format per 32 regles subdividides en 4 subgrups de 8 regles. Cadascun dels subgrups fa referència a cadascuna de les 4 coordenades cap on el mòdul vol realitzar el moviment (NE, SE, NW, SW). Cadascun d'aquests subgrups té 1 regla d'intencionalitat, 6 regles de verificació i 1 regla d'acció.

En aquest grup hem implementat les regles següents:

#### - Convex Transition



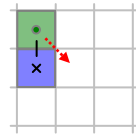


**- Moviment cap al Sud-Est (intenció):**

<b>Regla:</b>	Convex Transition SE 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
$Sfw\_A$ $E1,-1$ $A0001$ $!( (E0,1 E1,0) !(E0,1 F1,0 !( (T1,0, pfc\_*) !(T1,0, ****A) )) !(E1,0 F0,1 !( (T0,1, pfc\_*) !(T0,1, ****A) ))$ $!(F0,1 !( (T0,1, pfc\_*) !(T0,1, ****A) )) F1,0 !( (T1,0, pfc\_*) !(T1,0, ****A) ))$	
<b>Postcondicions:</b>	
$SfwSEA$	

**- Regla:** Convex Transition SE 1

Aquesta regla s'aplica als mòduls actius connectats al Sud que tenen possibilitats de moure's al Sud-Est i mostren aquesta intenció de moviment.

**- Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Convex Transition és potencialment més petita.

**- Precondicions:**

$Sfw\_A$  : afecta els mòduls actius de la fase [fw].

$E1,-1$  : afecta els mòduls que tenen un espai buit al Sud-Est.

$A0001$  : afecta els mòduls que estan connectats només al Sud.

$!( (E0,1 E1,0) !(E0,1 F1,0 !( (T1,0, pfc\_*) !(T1,0, ****A) )) !(E1,0 F0,1 !( (T0,1, pfc\_*) !(T0,1, ****A) )) !(F0,1 !( (T0,1, pfc\_*) !(T0,1, ****A) )) F1,0 !( (T1,0, pfc\_*) !(T1,0, ****A) ))$  : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

**- Postcondicions:**

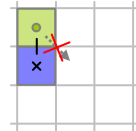
$SfwSEA$  : canvia el seu estat a "intenció de moviment al Sud-Est".

**- Moviment cap al Sud-Est (verificació):**

<b>Regla:</b>	Convex Transition SE 2 cant move
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	$Sfw_{SEA}$ $\neg(E1, -1 A0001)$
<b>Postcondicions:</b>	$Sfw\_A$

**- Regla:** Convex Transition SE 2 cant move

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.



**- Prioritat:** 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

**- Precondicions:**

$Sfw_{SEA}$  : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.

$\neg(E1, -1 A0001)$ : afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.

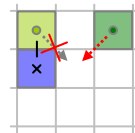
**- Postcondicions:**

$Sfw\_A$  : torna el seu estat a mode d'espera.

<b>Regla:</b>	Convex Transition SE 2 win NE
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSEA</i> <i>F2,0</i> <i>T2,0, fwSWA</i> <i>V2,0, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition SE 2 win NE

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwSEA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.

*F2,0* : afecta els mòduls que tenen un veí al Sud-Est Nord-Est.

*T2,0, fwSWA* : afecta els mòduls que tenen un veí al Sud-Est Nord-Est amb intenció de moure's a la mateixa posició.

*V2,0, C000 C000* : afecta els mòduls que tenen un veí al Sud-Est Nord-Est amb un valor potencial més petit.

- Postcondicions:

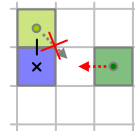
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Convex Transition SE 2 win E</i>
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSEA F2,-1 T2,-1, fwW_A V2,-1, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition SE 2 win E

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwSEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.*

*F2,-1 : afecta els mòduls que tenen un veí al Sud-Est Est.*

*T2,-1, fwW\_A : afecta els mòduls que tenen un veí al Sud-Est Est amb intenció de moure's a la mateixa posició.*

*V2,-1, C000 C000 : afecta els mòduls que tenen un veí al Sud-Est Est amb un valor potencial més petit.*

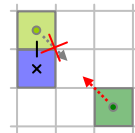
### - Postcondicions:

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition SE 2 win SE
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSEA</i> <i>F2,-2</i> <i>T2,-2, fwNWA</i> <i>V2,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition SE 2 win SE

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwSEA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.

*F2,-2* : afecta els mòduls que tenen un veí al Sud-Est Sud-Est.

*T2,-2, fwNWA* : afecta els mòduls que tenen un veí al Sud-Est Sud-Est amb intenció de moure's a la mateixa posició.

*V2,-2, C000 C000* : afecta els mòduls que tenen un veí al Sud-Est Sud-Est amb un valor potencial més petit.

- Postcondicions:

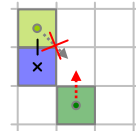
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Convex Transition SE 2 win S</i>
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSEA</i> <i>F1,-2</i> <i>T1,-2, fwN_A</i> <i>V1,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition SE 2 win S

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwSEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.*

*F1,-2 : afecta els mòduls que tenen un veí al Sud-Est Sud.*

*T1,-2, fwN\_A : afecta els mòduls que tenen un veí al Sud-Est Sud amb intenció de moure's a la mateixa posició.*

*V1,-2, C000 C000 : afecta els mòduls que tenen un veí al Sud-Est Sud amb un valor potencial més petit.*

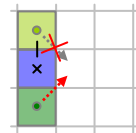
### - Postcondicions:

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition SE 2 win SW
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSEA</i> <i>F0,-2</i> <i>T0,-2, fwNEA</i> <i>V0,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition SE 2 win SW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwSEA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.

*F0,-2* : afecta els mòduls que tenen un veí al Sud-Est Sud-Oest.

*T0,-2, fwNEA* : afecta els mòduls que tenen un veí al Sud-Est Sud-Oest amb intenció de moure's a la mateixa posició.

*V0,-2, C000 C000* : afecta els mòduls que tenen un veí al Sud-Est Sud-Oest amb un valor potencial més petit.

- Postcondicions:

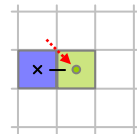
*Sfw\_\_A* : torna el seu estat a mode d'espera.

**- Moviment cap al Sud-Est (acció):**

<b>Regla:</b>	<i>Convex Transition SE 2 win</i>
<b>Prioritat:</b>	<i>8002</i>
<b>Precondicions:</b>	<i>SfwSEA</i>
<b>Postcondicions:</b>	<i>A0100 P1,-1 C001 - C001 0001 C020 + C020 0001 C021 - C021 0001 Sfw__A</i>

**- Regla:** *Convex Transition SE 2 win*

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Est i que res no els impedeix fer-ho.*



**- Prioritat:** *8002*

*La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.*

**- Precondicions:**

*SfwSEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Est.*

**- Postcondicions:**

*A0100 : es desconnecta del Sud i es connecta a l'Oest després de moure's.*

*P1,-1 : es mou una posició en direcció Sud-Est.*

*C001 - C001 0001 : Signe = Signe - 1*

*C020 + C020 0001 : Coordenada x = x + 1*

*C021 - C021 0001 : Coordenada y = y - 1*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

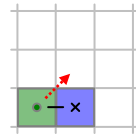


**- Moviment cap al Nord-Est (intenció):**

<b>Regla:</b>	Convex Transition NE 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
$Sfw\_A$ $E1,1$ $A0010$ $!( (E-1,0 E0,1 ) !(E-1,0 F0,1 !( (T0,1, pfc\_ * ) !(T0,1, ****A ))) !(E0,1 F-1,0 !( (T-1,0, pfc\_ * ) !(T-1,0, ****A )))$ $!(F-1,0 !( (T-1,0, pfc\_ * ) !(T-1,0, ****A ))) F0,1 !( (T0,1, pfc\_ * ) !(T0,1, ****A )))$	
<b>Postcondicions:</b>	
$SfwNEA$	

**- Regla:** Convex Transition NE 1

Aquesta regla s'aplica als mòduls actius connectats a l'Est que tenen possibilitats de moure's al Nord-Est i mostren aquesta intenció de moviment.

**- Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Convex Transition és potencialment més petita.

**- Precondicions:**

$Sfw\_A$  : afecta els mòduls actius de la fase [fw].

$E1,1$  : afecta els mòduls que tenen un espai buit al Nord-Est.

$A0010$  : afecta els mòduls que estan connectats només al Sud.

$!( (E-1,0 E0,1 ) !(E-1,0 F0,1 !( (T0,1, pfc\_ * ) !(T0,1, ****A ))) !(E0,1 F-1,0 !( (T-1,0, pfc\_ * ) !(T-1,0, ****A ))) !(F-1,0 !( (T-1,0, pfc\_ * ) !(T-1,0, ****A ))) F0,1 !( (T0,1, pfc\_ * ) !(T0,1, ****A )))$  : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

**- Postcondicions:**

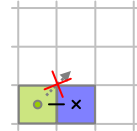
$SfwNEA$  : canvia el seu estat a "intenció de moviment al Nord-Est".

**- Moviment cap al Nord-Est (verificació):**

<b>Regla:</b>	<i>Convex Transition NE 2 cant move</i>
<b>Prioritat:</b>	<i>8003</i>
<b>Precondicions:</b>	<i>SfwNEA !(E1,1 A0010)</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

**- Regla:** Convex Transition NE 2 cant move

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.*



**- Prioritat:** 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

**- Precondicions:**

*SfwNEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.*

*!(E1,1 A0010): afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.*

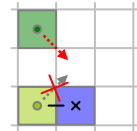
**- Postcondicions:**

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition NE 2 win NW
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNEA</i> <i>F0,2</i> <i>T0,2, fwSEA</i> <i>V0,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition NE 2 win NW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwNEA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.

*F0,2* : afecta els mòduls que tenen un veí al Nord-Est Nord-Oest.

*T0,2, fwSEA* : afecta els mòduls que tenen un veí al Nord-Est Nord-Oest amb intenció de moure's a la mateixa posició.

*V0,2, C000 C000* : afecta els mòduls que tenen un veí al Nord-Est Nord-Oest amb un valor potencial més petit.

- Postcondicions:

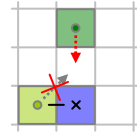
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Convex Transition NE 2 win N
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNEA</i> <i>F1,2</i> <i>T1,2, fwS_A</i> <i>V1,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition NE 2 win N

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwNEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.*

*F1,2 : afecta els mòduls que tenen un veí al Nord-Est Nord.*

*T1,2, fwS\_A : afecta els mòduls que tenen un veí al Nord-Est Nord amb intenció de moure's a la mateixa posició.*

*V1,2, C000 C000 : afecta els mòduls que tenen un veí al Nord-Est Nord amb un valor potencial més petit.*

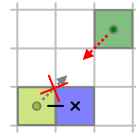
### - Postcondicions:

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition NE 2 win NE
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNEA</i> <i>F2,2</i> <i>T2,2, fwSWA</i> <i>V2,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition NE 2 win NE

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwNEA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.

*F2,2* : afecta els mòduls que tenen un veí al Nord-Est Nord-Est.

*T2,2, fwSWA* : afecta els mòduls que tenen un veí al Nord-Est Nord-Est amb intenció de moure's a la mateixa posició.

*V2,2, C000 C000* : afecta els mòduls que tenen un veí al Nord-Est Nord-Est amb un valor potencial més petit.

- Postcondicions:

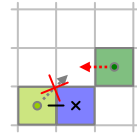
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Convex Transition NE 2 win E
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNEA</i> <i>F2,1</i> <i>T2,1, fwW_A</i> <i>V2,1, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition NE 2 win E

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.*



### - Prioritat: 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

### - Precondicions:

*SfwNEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.*

*F2,1 : afecta els mòduls que tenen un veí al Nord-Est Est.*

*T2,1, fwW\_A : afecta els mòduls que tenen un veí al Nord-Est Est amb intenció de moure's a la mateixa posició.*

*V2,1, C000 C000 : afecta els mòduls que tenen un veí al Nord-Est Est amb un valor potencial més petit.*

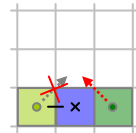
### - Postcondicions:

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition NE 2 win SE
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNEA</i> <i>F2,0</i> <i>T2,0, fwNWA</i> <i>V2,0, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition NE 2 win SE

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwNEA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.

*F2,0* : afecta els mòduls que tenen un veí al Nord-Est Sud-Est.

*T2,0, fwNWA* : afecta els mòduls que tenen un veí al Nord-Est Sud-Est amb intenció de moure's a la mateixa posició.

*V2,0, C000 C000* : afecta els mòduls que tenen un veí al Nord-Est Sud-Est amb un valor potencial més petit.

- Postcondicions:

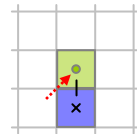
*Sfw\_\_A* : torna el seu estat a mode d'espera.

**- Moviment cap al Nord-Est (acció):**

<b>Regla:</b>	<i>Convex Transition NE 2 win</i>
<b>Prioritat:</b>	<i>8002</i>
<b>Precondicions:</b>	<i>SfwNEA</i>
<b>Postcondicions:</b>	<i>A0001 P1,1 C001 - C001 0001 C020 + C020 0001 C021 + C021 0001 Sfw__A</i>

**- Regla:** *Convex Transition NE 2 win*

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Est i que res no els impedeix fer-ho.*



**- Prioritat:** *8002*

*La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.*

**- Precondicions:**

*SfwNEA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Est.*

**- Postcondicions:**

*A0001 : es desconnecta de l'Est i es connecta al Sud després de moure's.*

*P1,1 : es mou una posició en direcció Nord-Est.*

*C001 - C001 0001 : Signe = Signe - 1*

*C020 + C020 0001 : Coordenada x = x + 1*

*C021 + C021 0001 : Coordenada y = y + 1*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

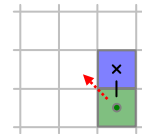


**- Moviment cap al Nord-Oest (intenció):**

<b>Regla:</b>	Convex Transition NW 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>E-1,1</i> <i>A1000</i> $!( (E0,-1 E-1,0) !(E0,-1 F-1,0 !( (T-1,0, pfc\_*) !(T-1,0, ****A) ))) !(E-1,0 F0,-1 !( (T0,-1, pfc\_*) !(T0,-1, ****A) )))$ $!(F0,-1 !( (T0,-1, pfc\_*) !(T0,-1, ****A) )) F-1,0 !( (T-1,0, pfc\_*) !(T-1,0, ****A) ))$	
<b>Postcondicions:</b>	
<i>SfwNWA</i>	

**- Regla:** Convex Transition NW 1

Aquesta regla s'aplica als mòduls actius connectats al Nord que tenen possibilitats de moure's al Nord-Oest i mostren aquesta intenció de moviment.

**- Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Convex Transition és potencialment més petita.

**- Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*E-1,1* : afecta els mòduls que tenen un espai buit al Nord-Oest.

*A1000* : afecta els mòduls que estan connectats només al Nord.

$!( (E0,-1 E-1,0) !(E0,-1 F-1,0 !( (T-1,0, pfc\_*) !(T-1,0, ****A) ))) !(E-1,0 F0,-1 !( (T0,-1, pfc\_*) !(T0,-1, ****A) ))) !(F0,-1 !( (T0,-1, pfc\_*) !(T0,-1, ****A) )) F-1,0 !( (T-1,0, pfc\_*) !(T-1,0, ****A) ))$  : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

**- Postcondicions:**

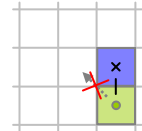
*SfwNWA* : canvia el seu estat a "intenció de moviment al Nord-Oest".

**- Moviment cap al Nord-Oest (verificació):**

<b>Regla:</b>	Convex Transition NW 2 cant move
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	$Sfw_{NWA}$ $!(E-1,1 A1000)$
<b>Postcondicions:</b>	$Sfw\_A$

**- Regla:** Convex Transition NW 2 cant move

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.*



**- Prioritat:** 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

**- Precondicions:**

$Sfw_{NWA}$  : *afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.*

$!(E-1,1 A1000)$ : *afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.*

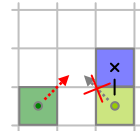
**- Postcondicions:**

$Sfw\_A$  : *torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition NW 2 win SW
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNWA</i> <i>F-2,0</i> <i>T-2,0, fwNEA</i> <i>V-2,0, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition NW 2 win SW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwNWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.

*F-2,0* : afecta els mòduls que tenen un veí al Nord-Oest Sud-Oest.

*T-2,0, fwNEA* : afecta els mòduls que tenen un veí al Nord-Oest Sud-Oest amb intenció de moure's a la mateixa posició.

*V-2,0, C000 C000* : afecta els mòduls que tenen un veí al Nord-Oest Sud-Oest amb un valor potencial més petit.

- Postcondicions:

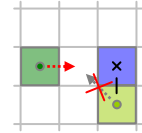
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Convex Transition NW 2 win W
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNWA</i> <i>F-2,1</i> <i>T-2,1, fwE_A</i> <i>V-2,1, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition NW 2 win W

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



### - Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwNWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.

*F-2,1* : afecta els mòduls que tenen un veí al Nord-Oest Oest.

*T-2,1, fwE\_A* : afecta els mòduls que tenen un veí al Nord-Oest Oest amb intenció de moure's a la mateixa posició.

*V-2,1, C000 C000* : afecta els mòduls que tenen un veí al Nord-Oest Oest amb un valor potencial més petit.

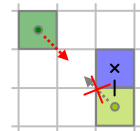
### - Postcondicions:

*Sfw\_\_A* : torna el seu estat a mode d'espera.

<b>Regla:</b>	Convex Transition NW 2 win NW
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNWA</i> <i>F-2,2</i> <i>T-2,2, fwSEA</i> <i>V-2,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition NW 2 win NW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwNWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.

*F-2,2* : afecta els mòduls que tenen un veí al Nord-Oest Nord-Oest.

*T-2,2, fwSEA* : afecta els mòduls que tenen un veí al Nord-Oest Nord-Oest amb intenció de moure's a la mateixa posició.

*V-2,2, C000 C000* : afecta els mòduls que tenen un veí al Nord-Oest Nord-Oest amb un valor potencial més petit.

- Postcondicions:

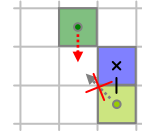
*Sfw\_\_A* : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Convex Transition NW 2 win N
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNWA</i> <i>F-1,2</i> <i>T-1,2, fwS_A</i> <i>V-1,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition NW 2 win N

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



### - Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwNWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.

*F-1,2* : afecta els mòduls que tenen un veí al Nord-Oest Nord.

*T-1,2, fwS\_A* : afecta els mòduls que tenen un veí al Nord-Oest Nord amb intenció de moure's a la mateixa posició.

*V-1,2, C000 C000* : afecta els mòduls que tenen un veí al Nord-Oest Nord amb un valor potencial més petit.

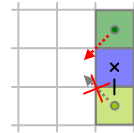
### - Postcondicions:

*Sfw\_\_A* : torna el seu estat a mode d'espera.

<b>Regla:</b>	Convex Transition NW 2 win NE
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwNWA</i> <i>F0,2</i> <i>T0,2, fwSWA</i> <i>V0,2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

- Regla: Convex Transition NW 2 win NE

Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwNWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.

*F0,2* : afecta els mòduls que tenen un veí al Nord-Oest Nord-Est.

*T0,2, fwSWA* : afecta els mòduls que tenen un veí al Nord-Oest Nord-Est amb intenció de moure's a la mateixa posició.

*V0,2, C000 C000* : afecta els mòduls que tenen un veí al Nord-Oest Nord-Est amb un valor potencial més petit.

- Postcondicions:

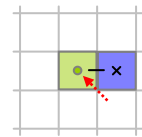
*Sfw\_\_A* : torna el seu estat a mode d'espera.

**- Moviment cap al Nord-Oest (acció):**

<b>Regla:</b>	<i>Convex Transition NW 2 win</i>
<b>Prioritat:</b>	<i>8002</i>
<b>Precondicions:</b>	<i>SfwNWA</i>
<b>Postcondicions:</b>	<i>A0010 P-1,1 C001 - C001 0001 C020 - C020 0001 C021 + C021 0001 Sfw__A</i>

**- Regla:** Convex Transition NW 2 win

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Nord-Oest i que res no els impedeix fer-ho.*



**- Prioritat:** 8002

*La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.*

**- Precondicions:**

*SfwNWA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Nord-Oest.*

**- Postcondicions:**

*A0010 : es desconnecta del Nord i es connecta a l'Est després de moure's.*

*P-1,1 : es mou una posició en direcció Nord-Oest.*

*C001 - C001 0001 : Signe = Signe - 1*

*C020 - C020 0001 : Coordenada x = x - 1*

*C021 + C021 0001 : Coordenada y = y + 1*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

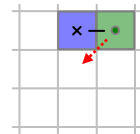


**- Moviment cap al Sud-Oest (intenció):**

<b>Regla:</b>	Convex Transition SW 1
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<p>Sfw__A</p> <p>E-1,-1</p> <p>A0100</p> <p><math>\neg ( \neg (E1,0 \ E0,-1) \ \neg (E1,0 \ F0,-1 \ \neg ( \neg (T0,-1, \ pfc\_*) \ \neg (T0,-1, \ ****A) ))) \ \neg (E0,-1 \ F1,0 \ \neg ( \neg (T1,0, \ pfc\_*) \ \neg (T1,0, \ ****A) )))</math></p> <p><math>\neg (F1,0 \ \neg ( \neg (T1,0, \ pfc\_*) \ \neg (T1,0, \ ****A) )) \ F0,-1 \ \neg ( \neg (T0,-1, \ pfc\_*) \ \neg (T0,-1, \ ****A) )))</math></p>
<b>Postcondicions:</b>	SfwSWA

**- Regla:** Convex Transition SW 1

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que tenen possibilitats de moure's al Sud-Oest i mostren aquesta intenció de moviment.

**- Prioritat:** 8003

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més petita que la dels grups de regles dels moviments Concave i Opposite perquè la distància que pot recórrer un mòdul amb Convex Transition és potencialment més petita.

**- Precondicions:**

Sfw\_\_A : afecta els mòduls actius de la fase [fw].

E-1,-1 : afecta els mòduls que tenen un espai buit al Sud-Oest.

A0100 : afecta els mòduls que estan connectats només al Nord.

$\neg ( \neg (E1,0 \ E0,-1) \ \neg (E1,0 \ F0,-1 \ \neg ( \neg (T0,-1, \ pfc\_*) \ \neg (T0,-1, \ ****A) ))) \ \neg (E0,-1 \ F1,0 \ \neg ( \neg (T1,0, \ pfc\_*) \ \neg (T1,0, \ ****A) ))) \ \neg (F1,0 \ \neg ( \neg (T1,0, \ pfc\_*) \ \neg (T1,0, \ ****A) )) \ F0,-1 \ \neg ( \neg (T0,-1, \ pfc\_*) \ \neg (T0,-1, \ ****A) )))$  : afecta els mòduls que han tingut la possibilitat d'analitzar totes les alternatives de moviments potencials.

**- Postcondicions:**

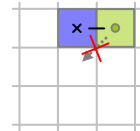
SfwSWA : canvia el seu estat a "intenció de moviment al Sud-Oest".

**- Moviment cap al Sud-Oest (verificació):**

<b>Regla:</b>	Convex Transition SW 2 cant move
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	$SfwSWA$ $!(E-1,-1 A0100)$
<b>Postcondicions:</b>	$Sfw\_A$

**- Regla:** Convex Transition SW 2 cant move

*Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest, que han deixat de tenir possibilitats de fer aquest moviment perquè alguna de les precondicions anteriors ha deixat de ser vàlida.*



**- Prioritat:** 8003

*La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

**- Precondicions:**

*SfwSWA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.*

*!(E-1,-1 A0100): afecta els mòduls que no poden moure's perquè alguna condició els ho impedeix.*

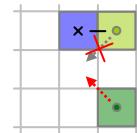
**- Postcondicions:**

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	Convex Transition SW 2 win SE
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<p>SfwSWA</p> <p>F0,-2</p> <p>T0,-2, fwNWA</p> <p>V0,-2, C000 C000</p>
<b>Postcondicions:</b>	Sfw__A

- Regla: Convex Transition SW 2 win SE

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

SfwSWA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.

F0,-2 : afecta els mòduls que tenen un veí al Sud-Oest Sud-Est.

T0,-2, fwNWA : afecta els mòduls que tenen un veí al Sud-Oest Sud-Est amb intenció de moure's a la mateixa posició.

V0,-2, C000 C000 : afecta els mòduls que tenen un veí al Sud-Oest Sud-Est amb un valor potencial més petit.

- Postcondicions:

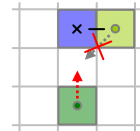
Sfw\_\_A : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Convex Transition SW 2 win S
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSWA</i> <i>F-1,-2</i> <i>T-1,-2, fwN_A</i> <i>V-1,-2, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition SW 2 win S

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



### - Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwSWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.

*F-1,-2* : afecta els mòduls que tenen un veí al Sud-Oest Sud.

*T-1,-2, fwN\_A* : afecta els mòduls que tenen un veí al Sud-Oest Sud amb intenció de moure's a la mateixa posició.

*V-1,-2, C000 C000* : afecta els mòduls que tenen un veí al Sud-Oest Sud amb un valor potencial més petit.

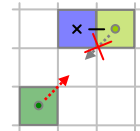
### - Postcondicions:

*Sfw\_\_A* : torna el seu estat a mode d'espera.

<b>Regla:</b>	Convex Transition SW 2 win SW
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<p>SfwSWA</p> <p>F-2,-2</p> <p>T-2,-2, fwNEA</p> <p>V-2,-2, C000 C000</p>
<b>Postcondicions:</b>	Sfw__A

- Regla: Convex Transition SW 2 win SW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

SfwSWA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.

F-2,-2 : afecta els mòduls que tenen un veí al Sud-Oest Sud-Oest.

T-2,-2, fwNEA : afecta els mòduls que tenen un veí al Sud-Oest Sud-Oest amb intenció de moure's a la mateixa posició.

V-2,-2, C000 C000 : afecta els mòduls que tenen un veí al Sud-Oest Sud-Oest amb un valor potencial més petit.

- Postcondicions:

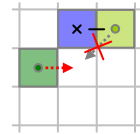
Sfw\_\_A : torna el seu estat a mode d'espera.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Convex Transition SW 2 win W
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<i>SfwSWA</i> <i>F-2,-1</i> <i>T-2,-1, fwE_A</i> <i>V-2,-1, C000 C000</i>
<b>Postcondicions:</b>	<i>Sfw__A</i>

### - Regla: Convex Transition SW 2 win W

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



### - Prioritat: 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

### - Precondicions:

*SfwSWA* : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.

*F-2,-1* : afecta els mòduls que tenen un veí al Sud-Oest Oest.

*T-2,-1, fwE\_A* : afecta els mòduls que tenen un veí al Sud-Oest Oest amb intenció de moure's a la mateixa posició.

*V-2,-1, C000 C000* : afecta els mòduls que tenen un veí al Sud-Oest Oest amb un valor potencial més petit.

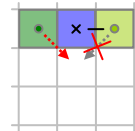
### - Postcondicions:

*Sfw\_\_A* : torna el seu estat a mode d'espera.

<b>Regla:</b>	Convex Transition SW 2 win NW
<b>Prioritat:</b>	8003
<b>Precondicions:</b>	<p>SfwSWA</p> <p>F-2,0</p> <p>T-2,0, fwSEA</p> <p>V-2,0, C000 C000</p>
<b>Postcondicions:</b>	Sfw__A

- **Regla:** Convex Transition SW 2 win NW

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest que no poden fer-ho perquè hi ha un altre mòdul amb valor potencial més petit que té preferència.



- **Prioritat:** 8003

La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- **Precondicions:**

SfwSWA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.

F-2,0 : afecta els mòduls que tenen un veí al Sud-Oest Nord-Oest.

T-2,0, fwSEA : afecta els mòduls que tenen un veí al Sud-Oest Nord-Oest amb intenció de moure's a la mateixa posició.

V-2,0, C000 C000 : afecta els mòduls que tenen un veí al Sud-Oest Nord-Oest amb un valor potencial més petit.

- **Postcondicions:**

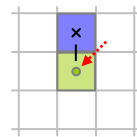
Sfw\_\_A : torna el seu estat a mode d'espera.

**- Moviment cap al Sud-Oest (acció):**

<b>Regla:</b>	Convex Transition SW 2 win
<b>Prioritat:</b>	8002
<b>Precondicions:</b>	SfwSWA
<b>Postcondicions:</b>	A1000 P-1,-1 C001 - C001 0001 C020 - C020 0001 C021 - C021 0001 Sfw__A

**- Regla:** Convex Transition SW 2 win

Aquesta regla s'aplica als mòduls amb intenció de moure's al Sud-Oest i que res no els impedeix fer-ho.



**- Prioritat:** 8002

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment anunciat prèviament.

**- Precondicions:**

SfwSWA : afecta els mòduls actius de la fase [fw] amb intenció de moure's al Sud-Oest.

**- Postcondicions:**

A1001 : es desconnecta de l'Oest i es connecta al Nord després de moure's.

P-1,-1 : es mou una posició en direcció Sud-Oest.

C001 - C001 0001 : Signe = Signe - 1

C020 - C020 0001 : Coordenada x = x - 1

C021 - C021 0001 : Coordenada y = y - 1

Sfw\_\_A : torna el seu estat a mode d'espera.

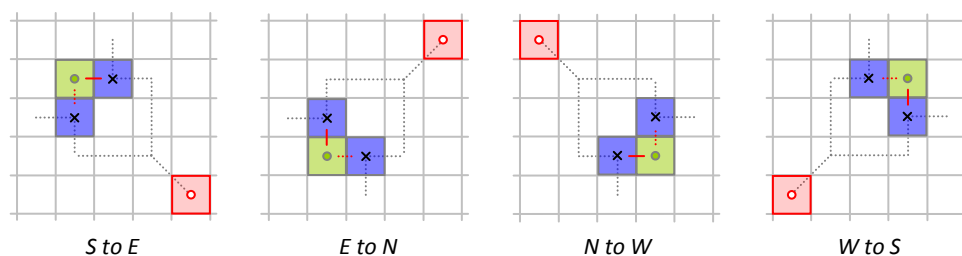


## C.4 Grup: "concave transition"

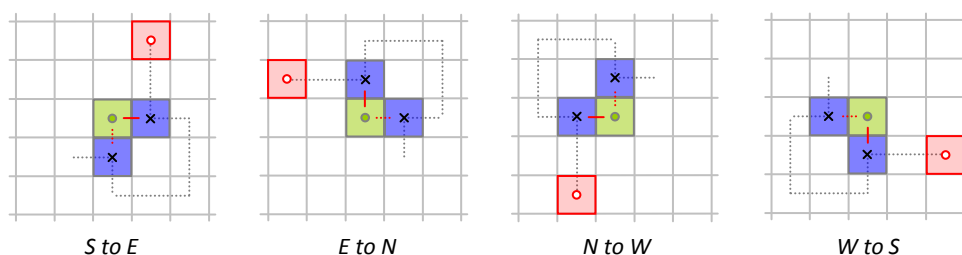
Aquest grup està format per 12 regles subdividides en 3 subgrups de 4 regles. El primer subgrup tracta els casos on els dos mòduls veïns implicats pertanyen a branques diferents. El segon i tercer subgrups tracten els casos on els veïns implicats formen part de la mateixa branca, un subgrup per als que millora el valor potencial, i un altre per als que empitjora. Cadascuna de les 4 regles d'un subgrup fa referència a una de les 4 coordenades a les quals el mòdul pot estar connectat (N, S, E, W).

En aquest grup hem implementat les regles següents:

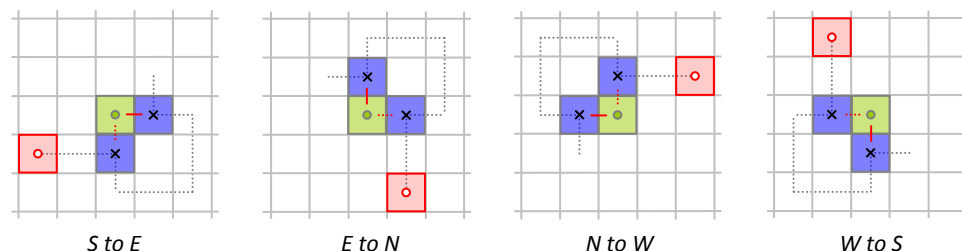
- Concave Transition diff branch



- Concave Transition same branch better Pot



- Concave Transition same branch worse Pot



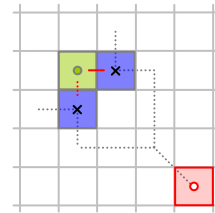
Nota: la posició del màster en aquestes figures no té significació geomètrica, sinó tan sols combinatòria i, per tant, sols contribueix a indicar les posicions relatives dels mòduls al llarg de l'arbre generador del robot.

- Branques diferents:

<b>Regla:</b>	Concave Transition S to E diff branch
<b>Prioritat:</b>	8007
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0001</i> <i>F1,0</i> <i>T1,0, pfc_*</i> <i>!T1,0, ****A</i> <i>V1,0, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A0010</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i>	

- Regla: Concave Transition S to E diff branch

Aquesta regla s'aplica als mòduls actius connectats al Sud que es connecten al veí de l'Est (que pertany a una branca diferent).



- Prioritat: 8007

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- Precondicions:

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0001* : afecta els mòduls que estan connectats només al Sud.

*F1,0* : afecta els mòduls que tenen un veí a l'Est.

*T1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *pfc\_*.

*!T1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Est en mode estàtic.

*V1,0, C004 C000* : afecta els mòduls que pertanyen a una branca diferent a la del mòdul de l'Est.

- Postcondicions:

*A0010* : es desconnecta del Sud i es connecta a l'Est.

*C000 + 1,0, C000 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0001* :  $Signe = 1$

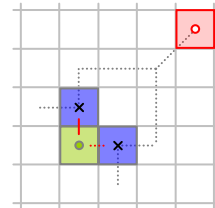
*C003 + 1,0, C003 0000* :  $Min = Min(E)$

*C004 + 1,0, C004 0000* :  $Max = Max(E)$

<b>Regla:</b>	Concave Transition E to N diff branch
<b>Prioritat:</b>	8007
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0010</i> <i>F0,1</i> <i>T0,1, pfc_*</i> <i>!T0,1, ****A</i> <i>V0,1, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A1000</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i>	

- **Regla:** Concave Transition E to N diff branch

Aquesta regla s'aplica als mòduls actius connectats a l'Est que es connecten al veí del Nord (que pertany a una branca diferent).



- **Prioritat:** 8007

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0010* : afecta els mòduls que estan connectats només a l'Est.

*F0,1* : afecta els mòduls que tenen un veí al Nord.

*T0,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord en estat de *pfc\_\**.

*!T0,1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Nord en mode estàtic.

*V0,1, C004 C000* : afecta els mòduls que pertanyen a una branca diferent a la del mòdul del Nord.

- **Postcondicions:**

*A1000* : es desconnecta de l'Est i es connecta al Nord.

*C000 + 0,1, C000 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,1, C003 0000* :  $Min = Min(N)$

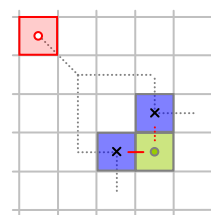
*C004 + 0,1, C004 0000* :  $Max = Max(N)$

### Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Concave Transition N to W diff branch</i>
<b>Prioritat:</b>	<i>8007</i>
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ***A</i> <i>V-1,0, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i>	

- Regla: Concave Transition N to W diff branch

*Aquesta regla s'aplica als mòduls actius connectats al Nord que es connecten al veí de l'Oest (que pertany a una branca diferent).*



- Prioritat: 8007

*La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.*

- Precondicions:

Sfw A : *afecta els mòduls actius de la fase [fw]*.

A1000 : afecta els mòduls que estan connectats només al Nord.

F-1,0 : *afecta els mòduls que tenen un veí a l'Oest.*

T-1,0, pfc \* : *afecta els mòduls que tenen el veí de l'Oest en estat de pfc.*

!T-1,0, \*\*\*\*A : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.

V-1,0, C004 C000 : *afecta els mòduls que pertanyen a una branca diferent a la del mòdul de l'Oest.*

- Postcondiciones:

A0100 : es desconnecta del Nord i es connecta a l'Oest.

C000 + -1,0, C000 0001 :  $V_{pot} = V_{pot}(W) + 1$

C001 + 0000 0001 : *Signe* = 1

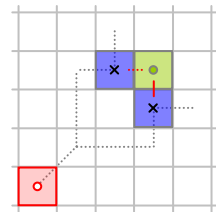
C003 + -1,0, C003 0000 :  $Min = Min(W)$

C004 + -1,0, C004 0000 :  $Max = Max(W)$

<b>Regla:</b>	Concave Transition W to S diff branch
<b>Prioritat:</b>	8007
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0100</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>V0,-1, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i>	

- Regla: Concave Transition W to S diff branch

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que es connecten al veí del Sud (que pertany a una branca diferent).



- Prioritat: 8007

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- Precondicions:

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*V0,-1, C004 C000* : afecta els mòduls que pertanyen a una branca diferent a la del mòdul del Sud.

- Postcondicions:

*A0001* : es desconnecta de l'Oest i es connecta al Sud.

*C000 + 0,-1, C000 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,-1, C003 0000* :  $Min = Min(S)$

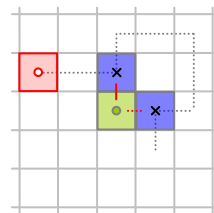
*C004 + 0,-1, C004 0000* :  $Max = Max(S)$



<b>Regla:</b>	Concave Transition E to N same branch better Pot
<b>Prioritat:</b>	8006
<b>Precondicions:</b>	
Sfw__A A0010 F0,1 T0,1, pfc_* !T0,1, ****A V0,1, C000 C000 V0,1, C003 C004 !V0,1, C004 C003 !( !W0,1, C002 0002 !V0,1, C002 0002 ) !(F1,1 !T1,1, ****A ))	
<b>Postcondicions:</b>	
A1000 C000 + 0,1, C000 0001 C001 + 0000 0001 C003 + 0,1, C003 0000 C004 + 0,1, C004 0000	

- **Regla:** Concave Transition E to N same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats a l'Est que es connecten al veí del Nord (que pertany a la mateixa branca i té un valor potencial més petit).



- **Prioritat:** 8006

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

Sfw\_\_A : afecta els mòduls actius de la fase [fw].

A0010 : afecta els mòduls que estan connectats només a l'Est.

F0,1 : afecta els mòduls que tenen un veí al Nord.

T0,1, pfc\_\* : afecta els mòduls que tenen el veí del Nord en estat de pfc\_.

!T0,1, \*\*\*\*A : afecta els mòduls que tenen el veí del Nord en mode estàtic.

V0,1, C000 C000 : afecta els mòduls que tenen pitjor valor potencial que el veí del Nord.

V0,1, C003 C004 !V0,1, C004 C003 : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Nord.

!( !W0,1, C002 0002 !V0,1, C002 0002 ) !(F1,1 !T1,1, \*\*\*\*A )) : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

A1000 : es desconnecta de l'Est i es connecta al Nord.

C000 + 0,1, C000 0001 :  $V_{pot} = V_{pot}(N) + 1$

C001 + 0000 0001 :  $Signe = 1$

C003 + 0,1, C003 0000 :  $Min = Min(N)$

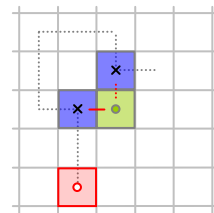
C004 + 0,1, C004 0000 :  $Max = Max(N)$

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Concave Transition N to W same branch better Pot
<b>Prioritat:</b>	8006
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ****A</i> <i>V-1,0, C000 C000</i> <i>V-1,0, C003 C004</i> <i>!V-1,0, C004 C003</i> <i>!( !(W-1,0, C002 0004 !V-1,0, C002 0004 ) !(F-1,1 !T-1,1, ****A ))</i>	
<b>Postcondicions:</b>	
<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i>	

- **Regla:** Concave Transition N to W same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats al Nord que es connecten al veí de l'Oest (que pertany a la mateixa branca i té un valor potencial més petit).



- **Prioritat:** 8006

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F-1,0* : afecta els mòduls que tenen un veí a l'Oest.

*T-1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *pfc\_\**.

*!T-1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.

*V-1,0, C000 C000* : afecta els mòduls que tenen pitjor valor potencial que el veí de l'Oest.

*V-1,0, C003 C004 !V-1,0, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Oest.

*!( !(W-1,0, C002 0004 !V-1,0, C002 0004 ) !(F-1,1 !T-1,1, \*\*\*\*A ))* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0100* : es desconnecta del Nord i es connecta a l'Oest.

*C000 + -1,0, C000 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + -1,0, C003 0000* :  $Min = Min(W)$

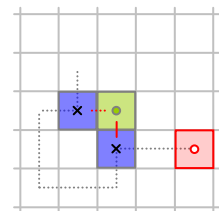
*C004 + -1,0, C004 0000* :  $Max = Max(W)$



<b>Regla:</b>	Concave Transition W to S same branch better Pot
<b>Prioritat:</b>	8006
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0100</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>V0,-1, C000 C000</i> <i>V0,-1, C003 C004</i> <i>!V0,-1, C004 C003</i> <i>!( !(W0,-1, C002 0003 !V0,-1, C002 0003 ) !(F-1,-1 !T-1,-1, ****A ))</i>	
<b>Postcondicions:</b>	
<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i>	

- **Regla:** Concave Transition W to S same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que es connecten al veí del Sud (que pertany a la mateixa branca i té un valor potencial més petit).



- **Prioritat:** 8006

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*V0,-1, C000 C000* : afecta els mòduls que tenen pitjor valor potencial que el veí del Sud.

*V0,-1, C003 C004 !V0,-1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Sud.

*!( !(W0,-1, C002 0003 !V0,-1, C002 0003 ) !(F-1,-1 !T-1,-1, \*\*\*\*A ))* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0001* : es desconnecta de l'Oest i es connecta al Sud.

*C000 + 0,-1, C000 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,-1, C003 0000* :  $Min = Min(S)$

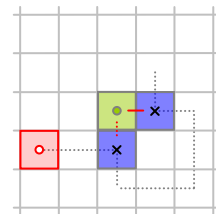
*C004 + 0,-1, C004 0000* :  $Max = Max(S)$

- **Mateixa branca amb empitjorament del valor potencial:**

<b>Regla:</b>	Concave Transition S to E same branch worse Pot
<b>Prioritat:</b>	8005
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0001</i> <i>F1,0</i> <i>T1,0, pfc_*</i> <i>!T1,0, ****A</i> <i>!V1,0, C000 C000</i> <i>V1,0, C003 C004</i> <i>!V1,0, C004 C003</i> <i>W0,-1, C002 0002 !V0,-1, C002 0002</i>	
<b>Postcondicions:</b>	
<i>A0010</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i>	

- **Regla:** Concave Transition S to E same branch worse Pot

Aquesta regla s'aplica als mòduls actius connectats al Sud que es connecten al veí de l'Est (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8005

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0001* : afecta els mòduls que estan connectats només al Sud.

*F1,0* : afecta els mòduls que tenen un veí a l'Est.

*T1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *pfc\_*.

*!T1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Est en mode estàtic.

*!V1,0, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí de l'Est.

*V1,0, C003 C004 !V1,0, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Est.

*W0,-1, C002 0002 !V0,-1, C002 0002* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0010* : es desconnecta del Sud i es connecta a l'Est.

*C000 + 1,0, C000 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0001* :  $Signe = 1$

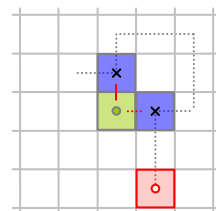
*C003 + 1,0, C003 0000* :  $Min = Min(E)$

*C004 + 1,0, C004 0000* :  $Max = Max(E)$

<b>Regla:</b>	Concave Transition E to N same branch worse Pot
<b>Prioritat:</b>	8005
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0010</i> <i>F0,1</i> <i>T0,1, pfc_*</i> <i>!T0,1, ****A</i> <i>!V0,1, C000 C000</i> <i>V0,1, C003 C004</i> <i>!V0,1, C004 C003</i> <i>W1,0, C002 0004 !V1,0, C002 0004</i>	
<b>Postcondicions:</b>	
<i>A1000</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i>	

- **Regla:** Concave Transition E to N same branch worse Pot

Aquesta regla s'aplica als mòduls actius connectats a l'Est que es connecten al veí del Nord (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8005

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0010* : afecta els mòduls que estan connectats només a l'Est.

*F0,1* : afecta els mòduls que tenen un veí al Nord.

*T0,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord en estat de pfc\_.

*!T0,1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Nord en mode estàtic.

*!V0,1, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí del Nord.

*V0,1, C003 C004 !V0,1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Nord.

*W1,0, C002 0004 !V1,0, C002 0004* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A1000* : es desconnecta de l'Est i es connecta al Nord.

*C000 + 0,1, C000 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,1, C003 0000* :  $Min = Min(N)$

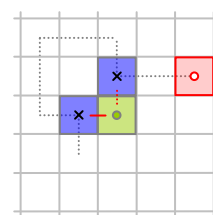
*C004 + 0,1, C004 0000* :  $Max = Max(N)$

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	Concave Transition N to W same branch worse Pot
<b>Prioritat:</b>	8005
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ****A</i> <i>!V-1,0, C000 C000</i> <i>V-1,0, C003 C004</i> <i>!V-1,0, C004 C003</i> <i>W0,1, C002 0003 !V0,1, C002 0003</i>	
<b>Postcondicions:</b>	
<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i>	

- **Regla:** Concave Transition N to W same branch worse Pot

Aquesta regla s'aplica als mòduls actius connectats al Nord que es connecten al veí de l'Oest (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8005

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F-1,0* : afecta els mòduls que tenen un veí a l'Oest.

*T-1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *pfc\_\**.

*!T-1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.

*!V-1,0, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí de l'Oest.

*V-1,0, C003 C004 !V-1,0, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Oest.

*W0,1, C002 0003 !V0,1, C002 0003* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0100* : es desconnecta del Nord i es connecta a l'Oest.

*C000 + -1,0, C000 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0001* :  $Signe = 1$

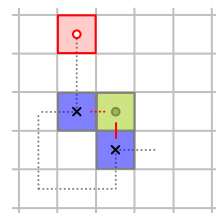
*C003 + -1,0, C003 0000* :  $Min = Min(W)$

*C004 + -1,0, C004 0000* :  $Max = Max(W)$

<b>Regla:</b>	Concave Transition W to S same branch worse Pot
<b>Prioritat:</b>	8005
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0100</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>!V0,-1, C000 C000</i> <i>V0,-1, C003 C004</i> <i>!V0,-1, C004 C003</i> <i>W-1,0, C002 0001 !V-1,0, C002 0001</i>	
<b>Postcondicions:</b>	
<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i>	

- **Regla:** Concave Transition W to S same branch worse Pot

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que es connecten al veí del Sud (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8005

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Concave Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de *pfc\_\**.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*!V0,-1, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí del Sud.

*V0,-1, C003 C004 !V0,-1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Sud.

*W-1,0, C002 0001 !V-1,0, C002 0001* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0001* : es desconnecta de l'Oest i es connecta al Sud.

*C000 + 0,-1, C000 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,-1, C003 0000* :  $Min = Min(S)$

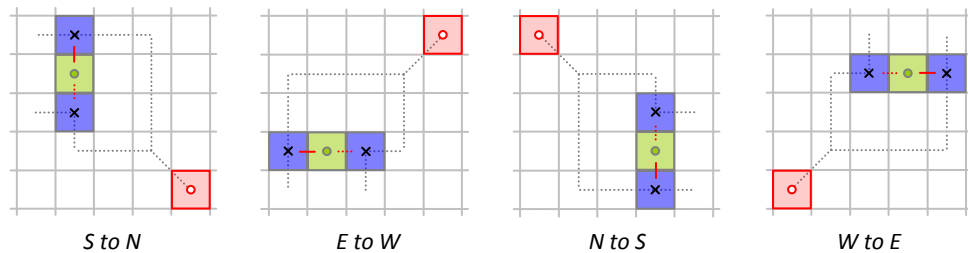
*C004 + 0,-1, C004 0000* :  $Max = Max(S)$

## C.5 Grup: “opposite transition”

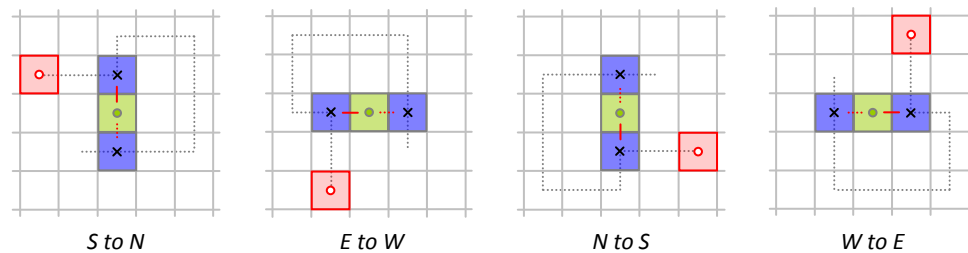
Aquest grup està format per 25 regles subdividides en 4 subgrups diferents, 3 d’ells de 4 regles i 1 amb les 13 restants. El primer subgrup tracta els casos on els dos mòduls veïns oposats pertanyen a branques diferents. El segon i tercer subgrups tracten els casos on aquests mòduls formen part de la mateixa branca, un subgrup per als que millora el valor potencial i un altre per als que empitjora. El quart subgrup tracta un cas especial en el qual un mòdul necessita aconseguir dades extra. Els tres primers subgrups tenen 4 regles on cadascuna fa referència a una de les 4 coordenades a les quals pot estar connectat el mòdul (N, S, E, W). El quart subgrup està format per 3 regles per cada coordenada (N, S, E, W) més una regla global.

En aquest grup hem implementat les regles següents:

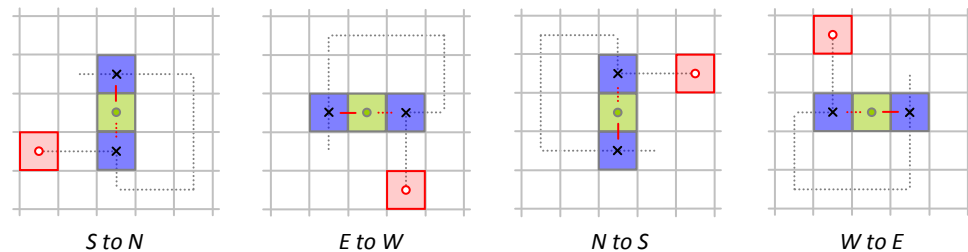
### - Opposite Transition diff branch



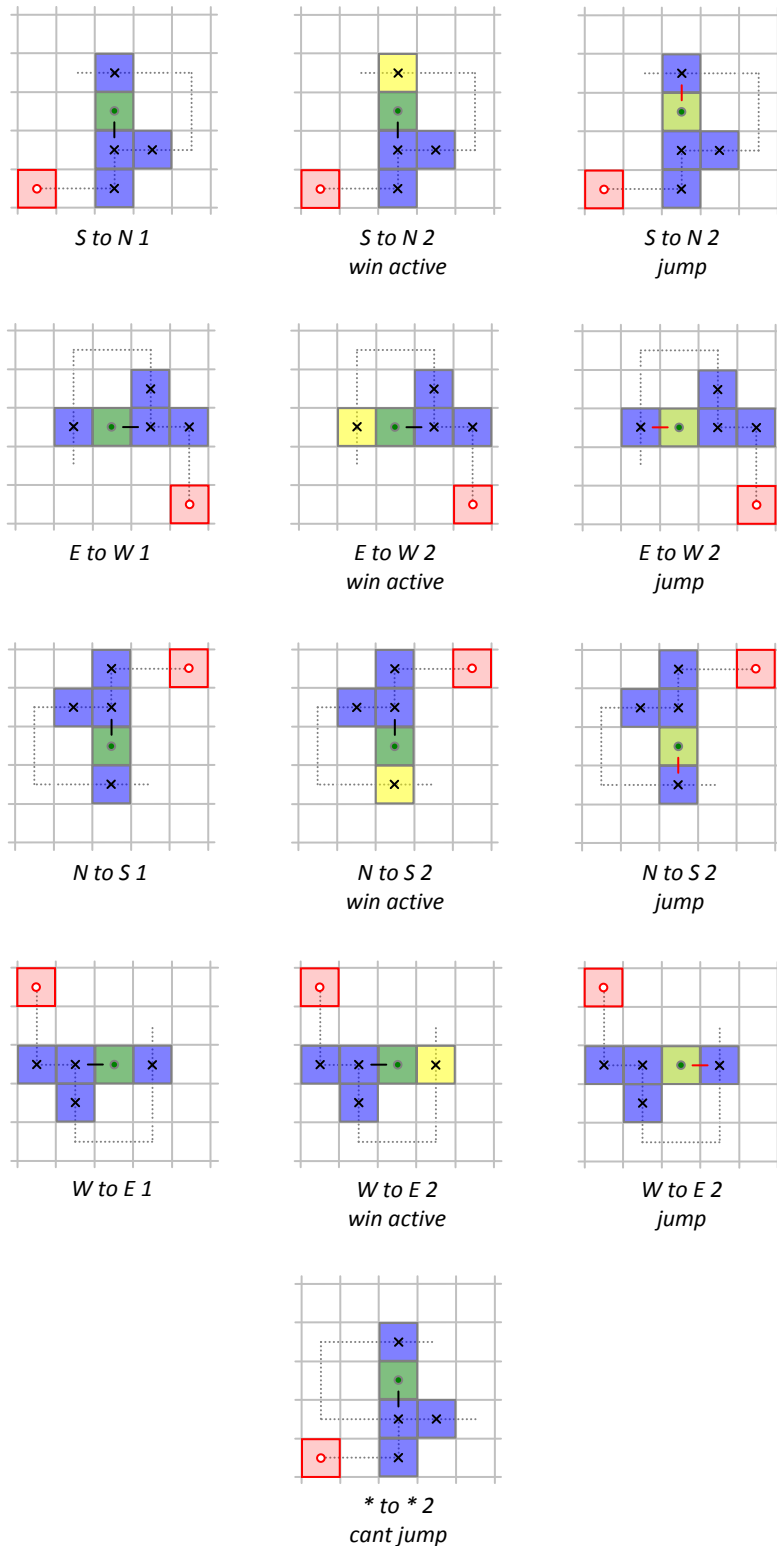
### - Opposite Transition same branch better Pot



### - Opposite Transition same branch worse Pot



- Opposite Transition same branch worse Pot (2 temps)



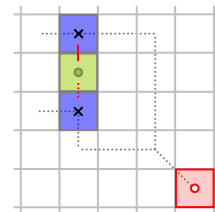
Nota: la posició del màster en aquestes figures no té significació geomètrica, sinó tan sols combinatòria i, per tant, sols contribueix a indicar les posicions relatives dels mòduls al llarg de l'arbre generador del robot.

- Branques diferents:

<b>Regla:</b>	<i>Opposite Transition S to N diff branch</i>
<b>Prioritat:</b>	8015
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0001</i> <i>F0,1</i> <i>T0,1, pfc_*</i> <i>!T0,1, ****A</i> <i>V0,1, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A1000</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i>	

- Regla: Opposite Transition S to N diff branch

*Aquesta regla s'aplica als mòduls actius connectats al Sud que es connecten al veí del Nord (que pertany a una branca diferent).*



- Prioritat: 8015

*La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.*

- Precondicions:

*Sfw\_\_A* : *afecta els mòduls actius de la fase [fw].*

*A0001* : *afecta els mòduls que estan connectats només al Sud.*

*F0,1* : *afecta els mòduls que tenen un veí al Nord.*

*T0,1, pfc\_\** : *afecta els mòduls que tenen el veí del Nord en estat de pfc\_.*

*!T0,1, \*\*\*\*A* : *afecta els mòduls que tenen el veí del Nord en mode estàtic.*

*V0,1, C004 C000* : *afecta els mòduls que pertanyen a una branca diferent a la del mòdul del Nord.*

- Postcondicions:

*A1000* : *es desconnecta del Sud i es connecta al Nord.*

*C000 + 0,1, C000 0001* : *Vpot = Vpot(N) + 1*

*C001 + 0000 0001* : *Signe = 1*

*C003 + 0,1, C003 0000* : *Min = Min(N)*

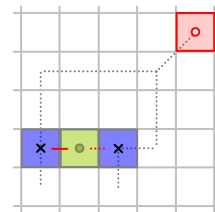
*C004 + 0,1, C004 0000* : *Max = Max(N)*



<b>Regla:</b>	<i>Opposite Transition E to W diff branch</i>
<b>Prioritat:</b>	8015
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0010</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ****A</i> <i>V-1,0, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i>	

- **Regla:** Opposite Transition E to W diff branch

*Aquesta regla s'aplica als mòduls actius connectats a l'Est que es connecten al veí de l'Oest (que pertany a una branca diferent).*



- **Prioritat:** 8015

*La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.*

- **Precondicions:**

*Sfw\_\_A : afecta els mòduls actius de la fase [fw].*

*A0010 : afecta els mòduls que estan connectats només a l'Est.*

*F-1,0 : afecta els mòduls que tenen un veí a l'Oest.*

*T-1,0, pfc\_\* : afecta els mòduls que tenen el veí de l'Oest en estat de pfc\_.*

*!T-1,0, \*\*\*\*A : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.*

*V-1,0, C004 C000 : afecta els mòduls que pertanyen a una branca diferent a la del mòdul de l'Oest.*

- **Postcondicions:**

*A0100 : es desconnecta de l'Est i es connecta a l'Oest.*

*C000 + -1,0, C000 0001 : Vpot = Vpot(W) + 1*

*C001 + 0000 0001 : Signe = 1*

*C003 + -1,0, C003 0000 : Min = Min(W)*

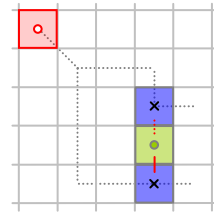
*C004 + -1,0, C004 0000 : Max = Max(W)*

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition N to S diff branch</i>
<b>Prioritat:</b>	8015
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>V0,-1, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i>	

- **Regla:** Opposite Transition N to S diff branch

Aquesta regla s'aplica als mòduls actius connectats al Nord que es connecten al veí del Sud (que pertany a una branca diferent).



- **Prioritat:** 8015

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*V0,-1, C004 C000* : afecta els mòduls que pertanyen a una branca diferent a la del mòdul del Sud.

- **Postcondicions:**

*A0001* : es desconnecta del Nord i es connecta al Sud.

*C000 + 0,-1, C000 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0001* :  $Signe = 1$

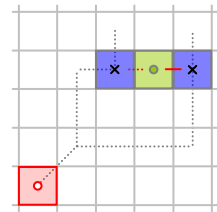
*C003 + 0,-1, C003 0000* :  $Min = Min(S)$

*C004 + 0,-1, C004 0000* :  $Max = Max(S)$

<b>Regla:</b>	<i>Opposite Transition W to E diff branch</i>
<b>Prioritat:</b>	8015
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0100</i> <i>F1,0</i> <i>T1,0, pfc_*</i> <i>!T1,0, ****A</i> <i>V1,0, C004 C000</i>	
<b>Postcondicions:</b>	
<i>A0010</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i>	

- **Regla:** Opposite Transition W to E diff branch

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que es connecten al veí de l'Est (que pertany a una branca diferent).



- **Prioritat:** 8015

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*F1,0* : afecta els mòduls que tenen un veí a l'Est.

*T1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *pfc\_\**.

*!T1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Est en mode estàtic.

*V1,0, C004 C000* : afecta els mòduls que pertanyen a una branca diferent a la del mòdul de l'Est.

- **Postcondicions:**

*A0010* : es desconnecta de l'Oest i es connecta a l'Est.

*C000 + 1,0, C000 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 1,0, C003 0000* :  $Min = Min(E)$

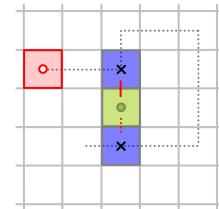
*C004 + 1,0, C004 0000* :  $Max = Max(E)$

**- Mateixa branca amb millora de valor potencial:**

<b>Regla:</b>	<i>Opposite Transition S to N same branch better Pot</i>
<b>Prioritat:</b>	8014
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0001</i> <i>F0,1</i> <i>T0,1, pfc_*</i> <i>!T0,1, ****A</i> <i>V0,1, C000 C000</i> <i>V0,1, C003 C004</i> <i>!V0,1, C004 C003</i> <i>!(</i> <i>!(W0,1, C002 0002 !V0,1, C002 0002 )</i> <i>!(W0,1, C002 0001 !V0,1, C002 0001 F1,1 !T1,1, ****A W1,1, C002 0002 !V1,1, C002 0002 V1,1, C003 C004 !V1,1, C004 C003 )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>A1000</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i>	

**- Regla:** Opposite Transition S to N same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats al Sud que es connecten al veí del Nord (que pertany a la mateixa branca i té un valor potencial més petit).



**- Prioritat:** 8014

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

**- Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0001* : afecta els mòduls que estan connectats només al Sud.

*F0,1* : afecta els mòduls que tenen un veí al Nord.

*T0,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord en estat de pfc\_.

*!T0,1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Nord en mode estàtic.

*V0,1, C000 C000* : afecta els mòduls que tenen pitjor valor potencial que el veí del Nord.

*V0,1, C003 C004 !V0,1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Nord.

*!( !(W0,1, C002 0002 !V0,1, C002 0002 ) !(W0,1, C002 0001 !V0,1, C002 0001 F1,1 !T1,1, \*\*\*\*A W1,1, C002 0002 !V1,1, C002 0002 V1,1, C003 C004 !V1,1, C004 C003 ))* : afecta els mòduls que es troben davant d'un cul de sac.

**- Postcondicions:**

*A1000* : es desconnecta del Sud i es connecta al Nord.

*C000 + 0,1, C000 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0001* :  $Signe = 1$

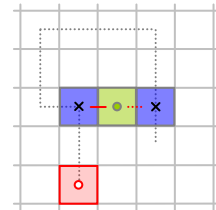
*C003 + 0,1, C003 0000* :  $Min = Min(N)$

*C004 + 0,1, C004 0000* :  $Max = Max(N)$

<b>Regla:</b>	<i>Opposite Transition E to W same branch better Pot</i>
<b>Prioritat:</b>	8014
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0010</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ****A</i> <i>V-1,0, C000 C000</i> <i>V-1,0, C003 C004</i> <i>!V-1,0, C004 C003</i> <i>!(</i> <i>!(W-1,0, C002 0004 !V-1,0, C002 0004 )</i> <i>!(W-1,0,C002 0002 !V-1,0,C002 0002 F-1,1 !T-1,1, ****A W-1,1,C002 0004 !V-1,1,C002 0004 V-1,1,C003 C004 !V-1,1,C004 C003)</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i>	

- **Regla:** Opposite Transition E to W same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats a l'Est que es connecten al veí de l'Oest (que pertany a la mateixa branca i té un valor potencial més petit).



- **Prioritat:** 8014

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0010* : afecta els mòduls que estan connectats només a l'Est.

*F-1,0* : afecta els mòduls que tenen un veí a l'Oest.

*T-1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de pfc\_.

*!T-1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.

*V-1,0, C000 C000* : afecta els mòduls que tenen pitjor valor potencial que el veí de l'Oest.

*V-1,0, C003 C004 !V-1,0, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Oest.

*!( !(W-1,0, C002 0004 !V-1,0, C002 0004 ) !(W-1,0,C002 0002 !V-1,0,C002 0002 F-1,1 !T-1,1, \*\*\*\*A W-1,1,C002 0004 !V-1,1,C002 0004 V-1,1,C003 C004 !V-1,1,C004 C003 ) )* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0100* : es desconnecta de l'Est i es connecta a l'Oest.

*C000 + -1,0, C000 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + -1,0, C003 0000* :  $Min = Min(W)$

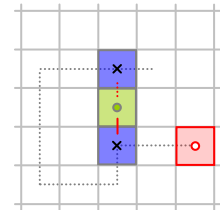
*C004 + -1,0, C004 0000* :  $Max = Max(W)$

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition N to S same branch better Pot</i>
<b>Prioritat:</b>	8014
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>V0,-1, C000 C000</i> <i>V0,-1, C003 C004</i> <i>!V0,-1, C004 C003</i> <i>!(</i> <i>!(W0,-1, C002 0003 !V0,-1, C002 0003 )</i> <i>!(W0,-1, C002 0004 !V0,-1, C002 0004 F-1,-1 !T-1,-1, ****A W-1,-1, C002 0003 !V-1,-1, C002 0003 V-1,-1, C003 C004 !V-1,-1, C004 C003 )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i>	

- **Regla:** Opposite Transition N to S same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats al Nord que es connecten al veí del Sud (que pertany a la mateixa branca i té un valor potencial més petit).



- **Prioritat:** 8014

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*V0,-1, C000 C000* : afecta els mòduls que tenen pitjor valor potencial que el veí del Sud.

*V0,-1, C003 C004 !V0,-1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Sud.

*!( ! (W0,-1, C002 0003 !V0,-1, C002 0003 ) ! (W0,-1, C002 0004 !V0,-1, C002 0004 F-1,-1, !T-1,-1, \*\*\*\*A W-1,-1, C002 0003 !V-1,-1, C002 0003 V-1,-1, C003 C004 !V-1,-1, C004 C003 ) )* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0001* : es desconnecta del Nord i es connecta al Sud.

*C000 + 0,-1, C000 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0001* :  $Signe = 1$

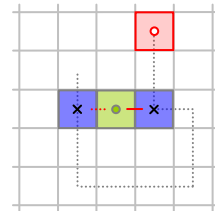
*C003 + 0,-1, C003 0000* :  $Min = Min(S)$

*C004 + 0,-1, C004 0000* :  $Max = Max(S)$

<b>Regla:</b>	<i>Opposite Transition W to E same branch better Pot</i>
<b>Prioritat:</b>	8014
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0100</i> <i>F1,0</i> <i>T1,0, pfc_*</i> <i>!T1,0, ****A</i> <i>V1,0, C000 C000</i> <i>V1,0, C003 C004</i> <i>!V1,0, C004 C003</i> <i>!(</i> <i>!(W1,0, C002 0001 !V1,0, C002 0001 )</i> <i>!(W1,0,C002 0003 !V1,0,C002 0003 F1,-1 !T1,-1, ****A W1,-1,C002 0001 !V1,-1,C002 0001 V1,-1,C003 C004 !V1,-1,C004 C003 )</i> <i>)</i>	
<b>Postcondicions:</b>	
<i>A0010</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i>	

- **Regla:** Opposite Transition W to E same branch better Pot

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que es connecten al veí de l'Est (que pertany a la mateixa branca i té un valor potencial més petit).



- **Prioritat:** 8014

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

*F1,0* : afecta els mòduls que tenen un veí a l'Est.

*T1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Est en estat de *pfc\_\**.

*!T1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Est en mode estàtic.

*V1,0, C000 C000* : afecta els mòduls que tenen pitjor valor potencial que el veí de l'Est.

*V1,0, C003 C004 !V1,0, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Est.

*!( !(W1,0, C002 0001 !V1,0, C002 0001 ) !(W1,0,C002 0003 !V1,0,C002 0003 F1,-1 !T1,-1, \*\*\*\*A W1,-1,C002 0001 !V1,-1,C002 0001 V1,-1,C003 C004 !V1,-1,C004 C003 ))* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0010* : es desconnecta de l'Oest i es connecta a l'Est.

*C000 + 1,0, C000 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0001* :  $Signe = 1$

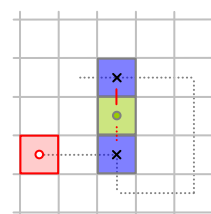
*C003 + 1,0, C003 0000* :  $Min = Min(E)$

*C004 + 1,0, C004 0000* :  $Max = Max(E)$

**- Mateixa branca amb empitjorament del valor potencial:**

<b>Regla:</b>	<i>Opposite Transition S to N same branch worse Pot</i>
<b>Prioritat:</b>	8013
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0001</i> <i>F0,1</i> <i>T0,1, pfc_*</i> <i>!T0,1, ****A</i> <i>!V0,1, C000 C000</i> <i>V0,1, C003 C004</i> <i>!V0,1, C004 C003</i> <i>W0,-1, C002 0002 !V0,-1, C002 0002</i>	
<b>Postcondicions:</b>	
<i>A1000</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i>	

- **Regla:** Opposite Transition S to N same branch worse Pot  
Aquesta regla s'aplica als mòduls actius connectats al Sud que es connecten al veí del Nord (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8013

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0001* : afecta els mòduls que estan connectats només al Sud.

*F0,1* : afecta els mòduls que tenen un veí al Nord.

*T0,1, pfc\_\** : afecta els mòduls que tenen el veí del Nord en estat de pfc\_.

*!T0,1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Nord en mode estàtic.

*!V0,1, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí del Nord.

*V0,1, C003 C004 !V0,1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Nord.

*W0,-1, C002 0002 !V0,-1, C002 0002* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A1000* : es desconnecta del Sud i es connecta al Nord.

*C000 + 0,1, C000 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,1, C003 0000* :  $Min = Min(N)$

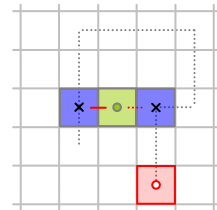
*C004 + 0,1, C004 0000* :  $Max = Max(N)$



<b>Regla:</b>	<i>Opposite Transition E to W same branch worse Pot</i>
<b>Prioritat:</b>	8013
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0010</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ****A</i> <i>!V-1,0, C000 C000</i> <i>V-1,0, C003 C004</i> <i>!V-1,0, C004 C003</i> <i>W1,0, C002 0004 !V1,0, C002 0004</i>	
<b>Postcondicions:</b>	
<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i>	

- **Regla:** Opposite Transition E to W same branch worse Pot

*Aquesta regla s'aplica als mòduls actius connectats a l'Est que es connecten al veí de l'Oest (que pertany a la mateixa branca i té un valor potencial més gran).*



- **Prioritat:** 8013

*La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.*

- **Precondicions:**

*Sfw\_\_A : afecta els mòduls actius de la fase [fw].*

*A0010 : afecta els mòduls que estan connectats només a l'Est.*

*F-1,0 : afecta els mòduls que tenen un veí a l'Oest.*

*T-1,0, pfc\_\* : afecta els mòduls que tenen el veí de l'Oest en estat de pfc\_.*

*!T-1,0, \*\*\*\*A : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.*

*!V-1,0, C000 C000 : afecta els mòduls que tenen millor valor potencial que el veí de l'Oest.*

*V-1,0, C003 C004 !V-1,0, C004 C003 : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Oest.*

*W1,0, C002 0004 !V1,0, C002 0004 : afecta els mòduls que es troben davant d'un cul de sac.*

- **Postcondicions:**

*A0100 : es desconnecta de l'Est i es connecta a l'Oest.*

*C000 + -1,0, C000 0001 :  $V_{pot} = V_{pot}(W) + 1$*

*C001 + 0000 0001 :  $Signe = 1$*

*C003 + -1,0, C003 0000 :  $Min = Min(W)$*

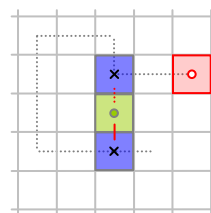
*C004 + -1,0, C004 0000 :  $Max = Max(W)$*

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition N to S same branch worse Pot</i>
<b>Prioritat:</b>	8013
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>!V0,-1, C000 C000</i> <i>V0,-1, C003 C004</i> <i>!V0,-1, C004 C003</i> <i>W0,1, C002 0003 !V0,1, C002 0003</i>	
<b>Postcondicions:</b>	
<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i>	

- **Regla:** Opposite Transition N to S same branch worse Pot

Aquesta regla s'aplica als mòduls actius connectats al Nord que es connecten al veí del Sud (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8013

La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*!V0,-1, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí del Sud.

*V0,-1, C003 C004 !V0,-1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Sud.

*W0,1, C002 0003 !V0,1, C002 0003* : afecta els mòduls que es troben davant d'un cul de sac.

- **Postcondicions:**

*A0001* : es desconnecta del Nord i es connecta al Sud.

*C000 + 0,-1, C000 0001* :  $V_{pot} = V_{pot}(S) + 1$

*C001 + 0000 0001* :  $Signe = 1$

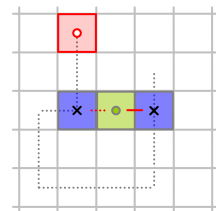
*C003 + 0,-1, C003 0000* :  $Min = Min(S)$

*C004 + 0,-1, C004 0000* :  $Max = Max(S)$

<b>Regla:</b>	<i>Opposite Transition W to E same branch worse Pot</i>
<b>Prioritat:</b>	8013
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0100</i> <i>F1,0</i> <i>T1,0, pfc_*</i> <i>!T1,0, ****A</i> <i>!V1,0, C000 C000</i> <i>V1,0, C003 C004</i> <i>!V1,0, C004 C003</i> <i>W-1,0, C002 0001 !V-1,0, C002 0001</i>	
<b>Postcondicions:</b>	
<i>A0010</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i>	

- **Regla:** Opposite Transition W to E same branch worse Pot

*Aquesta regla s'aplica als mòduls actius connectats a l'Oest que es connecten al veí de l'Est (que pertany a la mateixa branca i té un valor potencial més gran).*



- **Prioritat:** 8013

*La prioritat és més petita que la del grup de regles d'activació perquè un mòdul s'ha d'activar abans de poder-se moure. La prioritat és més gran que la dels grups de regles dels moviments Slide i Convex Transition perquè la distància que pot recórrer un mòdul amb Opposite Transition és potencialment més gran.*

- **Precondicions:**

*Sfw\_\_A : afecta els mòduls actius de la fase [fw].*

*A0100 : afecta els mòduls que estan connectats només a l'Oest.*

*F1,0 : afecta els mòduls que tenen un veí a l'Est.*

*T1,0, pfc\_\* : afecta els mòduls que tenen el veí de l'Est en estat de pfc\_.*

*!T1,0, \*\*\*\*A : afecta els mòduls que tenen el veí de l'Est en mode estàtic.*

*!V1,0, C000 C000 : afecta els mòduls que tenen millor valor potencial que el veí de l'Est.*

*V1,0, C003 C004 !V1,0, C004 C003 : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Est.*

*W-1,0, C002 0001 !V-1,0, C002 0001 : afecta els mòduls que es troben davant d'un cul de sac.*

- **Postcondicions:**

*A0010 : es desconnecta de l'Oest i es connecta a l'Est.*

*C000 + 1,0, C000 0001 :  $V_{pot} = V_{pot}(E) + 1$*

*C001 + 0000 0001 :  $Signe = 1$*

*C003 + 1,0, C003 0000 :  $Min = Min(E)$*

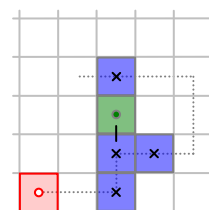
*C004 + 1,0, C004 0000 :  $Max = Max(E)$*

**- Mateixa branca amb empitjorament del valor potencial (2 temps) des del Sud:**

<b>Regla:</b>	<i>Opposite Transition S to N same branch worse Pot 1</i>
<b>Prioritat:</b>	8012
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0001</i> <i>F0,1</i> <i>T0,1, pfc_*</i> <i>!T0,1, ****A</i> <i>!V0,1, C000 C000</i> <i>V0,1, C003 C004</i> <i>!V0,1, C004 C003</i> <i>W0,-1, C002 0004 !V0,-1, C002 0004</i> <i>F1,-1</i> <i>!T1,-1, ****A</i> <i>W1,-1, C002 0002 !V1,-1, C002 0002</i>	
<b>Postcondicions:</b>	
<i>C010 + 0,1, C003 0000</i> <i>C011 + 0,1, C004 0000</i> <i>C012 + 1,-1, C003 0000</i> <i>C013 + 1,-1, C004 0000</i> <i>SfwdcA</i>	

- **Regla:** Opposite Transition S to N same branch worse Pot 1

*Aquesta regla s'aplica als mòduls actius connectats al Sud que tenen possibilitats de connectar-se al veí del Nord (que pertany a la mateixa branca i té un valor potencial més gran).*



- **Prioritat:** 8012

*La prioritat és més petita que la del grup de "Opposite Transition same branch worse Pot" (que tracta el cas general) perquè analitza el "cas especial" que necessita de 2 temps per resoldre quin moviment realitzar.*

- **Precondicions:**

*Sfw\_\_A : afecta els mòduls actius de la fase [fw].*

*A0001 : afecta els mòduls que estan connectats només al Sud.*

*F0,1 : afecta els mòduls que tenen un veí al Nord.*

*T0,1, pfc\_\* : afecta els mòduls que tenen el veí del Nord en estat de pfc\_.*

*!T0,1, \*\*\*\*A : afecta els mòduls que tenen el veí del Nord en mode estàtic.*

*!V0,1, C000 C000 : afecta els mòduls que tenen millor valor potencial que el veí del Nord.*

*V0,1, C003 C004 !V0,1, C004 C003 : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Nord.*

*W0,-1, C002 0004 !V0,-1, C002 0004 F1,-1 !T1,-1, \*\*\*\*A W1,-1, C002 0002 !V1,-1, C002 0002 : identifica el "cas especial".*

- **Postcondicions:**

*C010 + 0,1, C003 0000 : mínim del mòdul oposat = Min(N)*

*C011 + 0,1, C004 0000 : màxim del mòdul oposat = Max(N)*

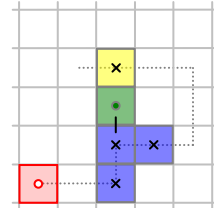
*C012 + 1,-1, C003 0000 : mínim de la subbranca = Min(SE)*

*C013 + 1,-1, C004 0000 : mínim de la subbranca = Min(SE)*

*SfwdcA : canvia el seu estat a "decidint camí".*

<b>Regla:</b>	<i>Opposite Transition S to N same branch worse Pot 2 win active</i>
<b>Prioritat:</b>	8011
<b>Precondicions:</b>	
<i>SfwdcA</i> <i>A0001</i> <i>!( !(T0,1, ****A ) !(T0,1, pfcA_ ))</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>C013 + 0000 0000</i> <i>Sfw__A</i>	

- **Regla:** Opposite Transition S to N same branch worse Pot 2 win active  
*Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite però que ja no poden fer-ho perquè el mòdul on s'avien de connectar es vol activar.*



- **Prioritat:** 8011

*La prioritat és més petita que la de les regles que capturen les dades "worse Pot 1" perquè, abans de verificar si pot o no realitzar l'acció, els mòduls han d'agafar les dades que necessiten. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

- **Precondicions:**

*SfwdcA : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".*

*A0001 : afecta els mòduls que estan connectats només al Sud.*

*!( !(T0,1, \*\*\*\*A ) !(T0,1, pfcA\_ )) : afecta els mòduls que tenen el veí del Nord activat o amb intenció d'activar-se.*

- **Postcondicions:**

*C010 + 0000 0000 : neteja el comptador.*

*C011 + 0000 0000 : neteja el comptador.*

*C012 + 0000 0000 : neteja el comptador.*

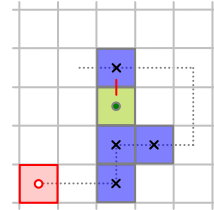
*C013 + 0000 0000 : neteja el comptador.*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition S to N same branch worse Pot 2 jump</i>
<b>Prioritat:</b>	8010
<b>Precondicions:</b>	<i>SfwdcA</i> <i>A0001</i>
<b>Postcondicions:</b>	<i>A1000</i> <i>C000 + 0,1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,1, C003 0000</i> <i>C004 + 0,1, C004 0000</i> <i>Sfw__A</i>

- **Regla:** Opposite Transition S to N same branch worse Pot 2 jump  
Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite i que res no els impedeix fer-ho.



- **Prioritat:** 8010

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment d'Opposite.

- **Precondicions:**

*SfwdcA* : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".

*A0001* : afecta els mòduls que estan connectats només al Sud.

- **Postcondicions:**

*A1000* : es desconnecta del Sud i es connecta al Nord.

*C000 + 0,1, C000 0001* :  $V_{pot} = V_{pot}(N) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 0,1, C003 0000* :  $Min = Min(N)$

*C004 + 0,1, C004 0000* :  $Max = Max(N)$

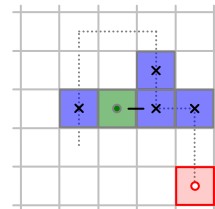
*Sfw\_\_A* : torna el seu estat a mode d'espera.

**- Mateixa branca amb empitjorament del valor potencial (2 temps) des de l'Est:**

<b>Regla:</b>	<i>Opposite Transition E to W same branch worse Pot 1</i>
<b>Prioritat:</b>	8012
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A0010</i> <i>F-1,0</i> <i>T-1,0, pfc_*</i> <i>!T-1,0, ****A</i> <i>!V-1,0, C000 C000</i> <i>V-1,0, C003 C004</i> <i>!V-1,0, C004 C003</i> <i>W1,0, C002 0003 !V1,0, C002 0003</i> <i>F1,1</i> <i>!T1,1, ****A</i> <i>W1,1, C002 0004 !V1,1, C002 0004</i>	
<b>Postcondicions:</b>	
<i>C010 + -1,0, C003 0000</i> <i>C011 + -1,0, C004 0000</i> <i>C012 + 1,1, C003 0000</i> <i>C013 + 1,1, C004 0000</i> <i>SfwdcA</i>	

**- Regla:** Opposite Transition E to W same branch worse Pot 1

Aquesta regla s'aplica als mòduls actius connectats a l'Est que tenen possibilitats de connectar-se al veí de l'Oest (que pertany a la mateixa branca i té un valor potencial més gran).



**- Prioritat:** 8012

La prioritat és més petita que la del grup de "Opposite Transition same branch worse Pot" (que tracta el cas general) perquè analitza el "cas especial" que necessita de 2 temps per resoldre quin moviment realitzar.

**- Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A0010* : afecta els mòduls que estan connectats només a l'Est.

*F-1,0* : afecta els mòduls que tenen un veí a l'Oest.

*T-1,0, pfc\_\** : afecta els mòduls que tenen el veí de l'Oest en estat de *pfc\_*.

*!T-1,0, \*\*\*\*A* : afecta els mòduls que tenen el veí de l'Oest en mode estàtic.

*!V-1,0, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí de l'Oest.

*V-1,0, C003 C004 !V-1,0, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul de l'Oest.

*W1,0, C002 0003 !V1,0, C002 0003 F1,1 !T1,1, \*\*\*\*A W1,1, C002 0004 !V1,1, C002 0004* : identifica el "cas especial".

**- Postcondicions:**

*C010 + -1,0, C003 0000* : mínim del mòdul oposat =  $\text{Min}(W)$

*C011 + -1,0, C004 0000* : màxim del mòdul oposat =  $\text{Max}(W)$

*C012 + 1,1, C003 0000* : mínim de la subbranca =  $\text{Min}(NE)$

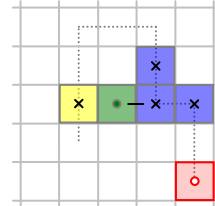
*C013 + 1,1, C004 0000* : mínim de la subbranca =  $\text{Min}(NE)$

*SfwdcA* : canvia el seu estat a "decidint camí".

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition E to W same branch worse Pot 2 win active</i>
<b>Prioritat:</b>	8011
<b>Precondicions:</b>	
<i>SfwdcA</i> <i>A0010</i> <i>!( !(T-1,0, ****A ) !(T-1,0, pfcA_ ))</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>C013 + 0000 0000</i> <i>Sfw__A</i>	

- **Regla:** Opposite Transition E to W same branch worse Pot 2 win active  
*Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite però que ja no poden fer-ho perquè el mòdul on s'avien de connectar es vol activar.*



- **Prioritat:** 8011

*La prioritat és més petita que la de les regles que capturen les dades "worse Pot 1" perquè, abans de verificar si pot o no realitzar l'acció, els mòduls han d'agafar les dades que necessiten. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

- **Precondicions:**

*SfwdcA : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".*

*A0010 : afecta els mòduls que estan connectats només a l'Est.*

*!( !(T-1,0, \*\*\*\*A ) !(T-1,0, pfcA\_ )) : afecta els mòduls que tenen el veí de l'Oest activat o amb intenció d'activar-se.*

- **Postcondicions:**

*C010 + 0000 0000 : neteja el comptador.*

*C011 + 0000 0000 : neteja el comptador.*

*C012 + 0000 0000 : neteja el comptador.*

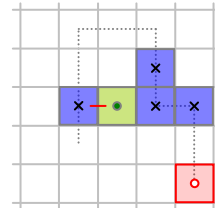
*C013 + 0000 0000 : neteja el comptador.*

*Sfw\_\_A : torna el seu estat a mode d'espera.*



<b>Regla:</b>	<i>Opposite Transition E to W same branch worse Pot 2 jump</i>
<b>Prioritat:</b>	8010
<b>Precondicions:</b>	<i>SfwdcA</i> <i>A0010</i>
<b>Postcondicions:</b>	<i>A0100</i> <i>C000 + -1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + -1,0, C003 0000</i> <i>C004 + -1,0, C004 0000</i> <i>Sfw__A</i>

- **Regla:** Opposite Transition E to W same branch worse Pot 2 jump  
Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite i que res no els impedeix fer-ho.



- **Prioritat:** 8010

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment d'Opposite.

- **Precondicions:**

*SfwdcA* : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".

*A0010* : afecta els mòduls que estan connectats només a l'Est.

- **Postcondicions:**

*A0100* : es desconnecta de l'Est i es connecta a l'Oest.

*C000 + -1,0, C000 0001* :  $V_{pot} = V_{pot}(W) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + -1,0, C003 0000* :  $Min = Min(W)$

*C004 + -1,0, C004 0000* :  $Max = Max(W)$

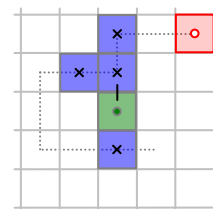
*Sfw\_\_A* : torna el seu estat a mode d'espera.

**- Mateixa branca amb empitjorament del valor potencial (2 temps) des del Nord:**

<b>Regla:</b>	<i>Opposite Transition N to S same branch worse Pot 1</i>
<b>Prioritat:</b>	8012
<b>Precondicions:</b>	
<i>Sfw__A</i> <i>A1000</i> <i>F0,-1</i> <i>T0,-1, pfc_*</i> <i>!T0,-1, ****A</i> <i>!V0,-1, C000 C000</i> <i>V0,-1, C003 C004</i> <i>!V0,-1, C004 C003</i> <i>W0,1, C002 0001 !V0,1, C002 0001</i> <i>F-1,1</i> <i>!T-1,1, ****A</i> <i>W-1,1, C002 0003 !V-1,1, C002 0003</i>	
<b>Postcondicions:</b>	
<i>C010 + 0,-1, C003 0000</i> <i>C011 + 0,-1, C004 0000</i> <i>C012 + -1,1, C003 0000</i> <i>C013 + -1,1, C004 0000</i> <i>SfwdcA</i>	

- **Regla:** Opposite Transition N to S same branch worse Pot 1

Aquesta regla s'aplica als mòduls actius connectats al Nord que tenen possibilitats de connectar-se al veí del Sud (que pertany a la mateixa branca i té un valor potencial més gran).



- **Prioritat:** 8012

La prioritat és més petita que la del grup de "Opposite Transition same branch worse Pot" (que tracta el cas general) perquè analitza el "cas especial" que necessita de 2 temps per resoldre quin moviment realitzar.

- **Precondicions:**

*Sfw\_\_A* : afecta els mòduls actius de la fase [fw].

*A1000* : afecta els mòduls que estan connectats només al Nord.

*F0,-1* : afecta els mòduls que tenen un veí al Sud.

*T0,-1, pfc\_\** : afecta els mòduls que tenen el veí del Sud en estat de pfc\_.

*!T0,-1, \*\*\*\*A* : afecta els mòduls que tenen el veí del Sud en mode estàtic.

*!V0,-1, C000 C000* : afecta els mòduls que tenen millor valor potencial que el veí del Sud.

*V0,-1, C003 C004 !V0,-1, C004 C003* : afecta els mòduls que pertanyen a la mateixa branca que la del mòdul del Sud.

*W0,1, C002 0001 !V0,1, C002 0001 F-1,1 !T-1,1, \*\*\*\*A W-1,1, C002 0003 !V-1,1, C002 0003* : identifica el "cas especial".

- **Postcondicions:**

*C010 + 0,-1, C003 0000* : mínim del mòdul oposat = Min(S)

*C011 + 0,-1, C004 0000* : màxim del mòdul oposat = Max(S)

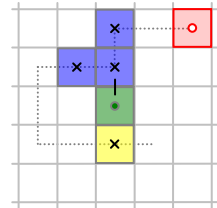
*C012 + -1,1, C003 0000* : mínim de la subbranca = Min(NW)

*C013 + -1,1, C004 0000* : mínim de la subbranca = Min(NW)

*SfwdcA* : canvia el seu estat a "decidint camí".

<b>Regla:</b>	<i>Opposite Transition N to S same branch worse Pot 2 win active</i>
<b>Prioritat:</b>	8011
<b>Precondicions:</b>	
<i>SfwdcA</i> <i>A1000</i> <i>!( !(T0,-1, ****A ) !(T0,-1, pfcA_ ))</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>C013 + 0000 0000</i> <i>Sfw__A</i>	

- **Regla:** Opposite Transition N to S same branch worse Pot 2 win active  
*Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite però que ja no poden fer-ho perquè el mòdul on s'avien de connectar es vol activar.*



- **Prioritat:** 8011

*La prioritat és més petita que la de les regles que capturen les dades "worse Pot 1" perquè, abans de verificar si pot o no realitzar l'acció, els mòduls han d'agafar les dades que necessiten. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

- **Precondicions:**

*SfwdcA : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".*

*A1000 : afecta els mòduls que estan connectats només al Nord.*

*!( !(T0,-1, \*\*\*\*A ) !(T0,-1, pfcA\_ )) : afecta els mòduls que tenen el veí del Sud activat o amb intenció d'activar-se.*

- **Postcondicions:**

*C010 + 0000 0000 : neteja el comptador.*

*C011 + 0000 0000 : neteja el comptador.*

*C012 + 0000 0000 : neteja el comptador.*

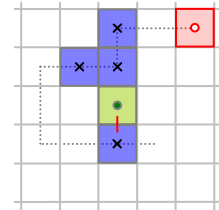
*C013 + 0000 0000 : neteja el comptador.*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition N to S same branch worse Pot 2 jump</i>
<b>Prioritat:</b>	8010
<b>Precondicions:</b>	<i>SfwdcA</i> <i>A1000</i>
<b>Postcondicions:</b>	<i>A0001</i> <i>C000 + 0,-1, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 0,-1, C003 0000</i> <i>C004 + 0,-1, C004 0000</i> <i>Sfw__A</i>

- **Regla:** Opposite Transition N to S same branch worse Pot 2 jump  
*Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite i que res no els impedeix fer-ho.*



- **Prioritat:** 8010

*La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment d'Opposite.*

- **Precondicions:**

*SfwdcA : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".*

*A1000 : afecta els mòduls que estan connectats només al Nord.*

- **Postcondicions:**

*A0001 : es desconnecta del Nord i es connecta al Sud.*

*C000 + 0,-1, C000 0001 : Vpot = Vpot(S) + 1*

*C001 + 0000 0001 : Signe = 1*

*C003 + 0,-1, C003 0000 : Min = Min(S)*

*C004 + 0,-1, C004 0000 : Max = Max(S)*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

- Regla: Opposite Transition W to E same branch worse Pot 1

*Aquesta regla s'aplica als mòduls actius connectats a l'Oest que tenen possibilitats de connectar-se al veí de l'Est (que pertany a la mateixa branca i té un valor potencial més gran).*




Diagrama d'un module de 3x3:

- Cella superior esquerra (0,0): Quadrat vermell buit.
- Cella superior mitjana (0,1): Quadrat blau amb una 'x' blanca.
- Cella superior dreta (0,2): Quadrat blau amb una 'x' blanca.
- Cella mitjana esquerra (1,0): Quadrat blau amb una 'x' blanca.
- Cella mitjana mitjana (1,1): Quadrat verd amb un punt negre.
- Cella mitjana dreta (1,2): Quadrat blau amb una 'x' blanca.
- Cella inferior esquerra (2,0): Quadrat blau amb una 'x' blanca.
- Cella inferior mitjana (2,1): Quadrat blau buit.
- Cella inferior dreta (2,2): Quadrat blau buit.

Una línia puntejada connecta la cella superior esquerra amb la cella inferior dreta.

Aquesta regla s'aplica als mòduls actius connectats a l'Oest que

*La prioritat és més petita que la del grup de "Opposite Transition same branch worse"*

Sfw A : *afecta els mòduls actius de la fase [fw]*.

A0100 : *afecta els mòduls que estan connectats només a l'Oest.*

F1,0 : *afecta els mòduls que tenen un veí a l'Est.*

T1,0, pfc\_\* : *afecta els mòduls que tenen el veí de l'Est en estat de pfc\_.*

!T1,0, \*\*\*\*A : *afecta els mòduls que tenen el veí de l'Est en mode estàtic.*

!V1,0, C000 C000 : *afecta els mòduls que tenen millor valor potencial que el veí de l'Est.*

V1,0, C003 C004 !V1,0, C004 C003 : *afecta els mòduls que pertanyen a la mateixa*

W-1,0, C002 0002 !V-1,0, C002 0002 F-1,-1 !T-1,-1, \*\*\*\*A W-1,-1, C002 0001 !V-1,-1,

C010 + 1,0, C003 0000 : *mínim del mòdul oposat* =  $Min(E)$

C011 + 1,0, C004 0000 : *màxim del mòdul oposat* =  $Max(E)$

C012 + -1,-1, C003 0000 : *mínim de la subbranca* = *Min(SW)*

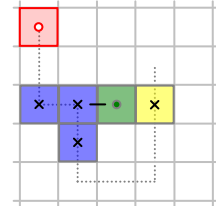
C013 + -1,-1, C004 0000 : *mínim de la subbranca* = *Min(SW)*

SfwdcA : *canvia el seu estat a “decidint camí”*.

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Opposite Transition W to E same branch worse Pot 2 win active</i>
<b>Prioritat:</b>	8011
<b>Precondicions:</b>	
<i>SfwdcA</i> <i>A0100</i> <i>!( !(T1,0, ****A ) !(T1,0, pfcA_ ))</i>	
<b>Postcondicions:</b>	
<i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>C013 + 0000 0000</i> <i>Sfw__A</i>	

- **Regla:** Opposite Transition W to E same branch worse Pot 2 win active  
*Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite però que ja no poden fer-ho perquè el mòdul on s'avien de connectar es vol activar.*



- **Prioritat:** 8011

*La prioritat és més petita que la de les regles que capturen les dades "worse Pot 1" perquè, abans de verificar si pot o no realitzar l'acció, els mòduls han d'agafar les dades que necessiten. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.*

- **Precondicions:**

*SfwdcA : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".*

*A0100 : afecta els mòduls que estan connectats només a l'Oest.*

*!( !(T1,0, \*\*\*\*A ) !(T1,0, pfcA\_ )) : afecta els mòduls que tenen el veí de l'Est activat o amb intenció d'activar-se.*

- **Postcondicions:**

*C010 + 0000 0000 : neteja el comptador.*

*C011 + 0000 0000 : neteja el comptador.*

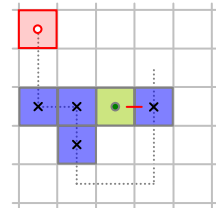
*C012 + 0000 0000 : neteja el comptador.*

*C013 + 0000 0000 : neteja el comptador.*

*Sfw\_\_A : torna el seu estat a mode d'espera.*

<b>Regla:</b>	<i>Opposite Transition W to E same branch worse Pot 2 jump</i>
<b>Prioritat:</b>	8010
<b>Precondicions:</b>	
<i>SfwdcA</i> <i>A0100</i>	
<b>Postcondicions:</b>	
<i>A0010</i> <i>C000 + 1,0, C000 0001</i> <i>C001 + 0000 0001</i> <i>C003 + 1,0, C003 0000</i> <i>C004 + 1,0, C004 0000</i> <i>Sfw__A</i>	

- **Regla:** Opposite Transition W to E same branch worse Pot 2 jump  
Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite i que res no els impedeix fer-ho.



- **Prioritat:** 8010

La prioritat és més petita que la de les regles de verificació perquè en cas que cap regla de verificació ho impedeixi, el mòdul realitza el moviment d'Opposite.

- **Precondicions:**

*SfwdcA* : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".

*A0100* : afecta els mòduls que estan connectats només a l'Oest.

- **Postcondicions:**

*A0010* : es desconnecta de l'Oest i es connecta a l'Est.

*C000 + 1,0, C000 0001* :  $V_{pot} = V_{pot}(E) + 1$

*C001 + 0000 0001* :  $Signe = 1$

*C003 + 1,0, C003 0000* :  $Min = Min(E)$

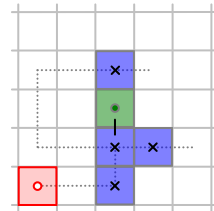
*C004 + 1,0, C004 0000* :  $Max = Max(E)$

*Sfw\_\_A* : torna el seu estat a mode d'espera.

**- Mateixa branca amb empitjorament del valor potencial (2 temps) verificació global:**

<b>Regla:</b>	<i>Opposite Transition * to * same branch worse Pot 2 cant jump</i>
<b>Prioritat:</b>	8011
<b>Precondicions:</b>	<i>SfwdcA</i> <i>!(&lt;C010 C013 !&lt;C011 C012 )</i>
<b>Postcondicions:</b>	<i>C010 + 0000 0000</i> <i>C011 + 0000 0000</i> <i>C012 + 0000 0000</i> <i>C013 + 0000 0000</i> <i>Sfw__A</i>

- Regla: Opposite Transition \* to \* same branch worse Pot 2 cant jump  
Aquesta regla s'aplica als mòduls que estan decidint si poden fer Opposite però que no el fan perquè la forma de la branca indica que el mòdul no està entrant al cul de sac.



- Prioritat: 8011

La prioritat és més petita que la de les regles que capturen les dades "worse Pot 1" perquè, abans de verificar si pot o no realitzar l'acció, els mòduls han d'agafar les dades que necessiten. La prioritat és la mateixa que la de la resta de regles de verificació perquè totes les comprovacions es fan simultàniament.

- Precondicions:

*SfwdcA* : afecta els mòduls actius de la fase [fw] en estat de "decidint camí".

*!(<C010 C013 !<C011 C012 )* : afecta els mòduls que, finalment, estan sortint d'un cul de sac.

- Postcondicions:

*C010 + 0000 0000* : neteja el comptador.

*C011 + 0000 0000* : neteja el comptador.

*C012 + 0000 0000* : neteja el comptador.

*C013 + 0000 0000* : neteja el comptador.

*Sfw\_\_A* : torna el seu estat a mode d'espera.

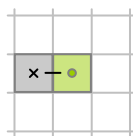


## C.6 Grup: "strip"

Aquest grup està format per 3 regles de 2 tipus, finalització i marcatge. Una regla per finalitzar la fase adherint-se a la tira, i dues per marcar l'últim mòdul actiu.

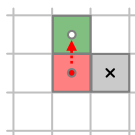
En aquest grup hem implementat les regles següents:

- Strip

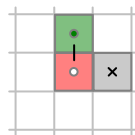


*Complete*

- Mark Leaf Strip



*Send*



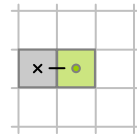
*Receive*

## Annex C - Regles de la fase de reconfiguració a la forma canònica

<b>Regla:</b>	<i>Complete Strip</i>
<b>Prioritat:</b>	8020
<b>Precondicions:</b>	
<i>Sfw__*</i> <i>!( !T-1,0, ****M !T-1,0, pfc_E )</i>	
<b>Postcondicions:</b>	
<i>Spfc_E</i> <i>C000 + 0000 0000</i> <i>C001 + 0000 0000</i> <i>C002 + 0000 0000</i> <i>C003 + 0000 0000</i> <i>C004 + 0000 0000</i>	

### - Regla: Complete Strip

Aquesta regla s'aplica als mòduls que arriben al final de la tira per connectar-s'hi.



### - Prioritat: 8020

La prioritat és més gran que la de les regles de moviment per evitar que un cop arriba el mòdul al seu destí no es continui movent.

### - Precondicions:

*Sfw\_\_\** : afecta a qualsevol mòdul de la fase [fw] en mode d'espera.

*!( !T-1,0, \*\*\*\*M !T-1,0, pfc\_E )* : afecta els mòduls que tenen a l'Oest el màster o un mòdul que pertany a la tira.

### - Postcondicions:

*Sfw\_\_A* : torna el seu estat a mode d'espera.

*Spfc\_E* : canvia el seu estat a "End" (pertanyent a la tira).

*C000 + 0000 0000* : neteja el comptador.

*C001 + 0000 0000* : neteja el comptador.

*C002 + 0000 0000* : neteja el comptador.

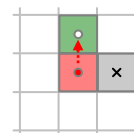
*C003 + 0000 0000* : neteja el comptador.

*C004 + 0000 0000* : neteja el comptador.

<b>Regla:</b>	<i>Send Mark Leaf Strip</i>
<b>Prioritat:</b>	8020
<b>Precondicions:</b>	
<i>Spfc_M</i> <i>A10**</i> <i>T0,1, fw__A</i> <i>W0,1, C024 0000</i>	
<b>Postcondicions:</b>	
<i>MN LStrp</i>	

- **Regla:** Send Mark Leaf Strip

Aquesta regla s'aplica al màster quan té l'últim mòdul actiu al Nord per enviar-li la marca de líder de la tira.



- **Prioritat:** 8020

La prioritat és més gran que la de les regles de moviment per evitar que el mòdul es mogui abans de gestionar la marca de líder de la tira.

- **Precondicions:**

*Spfc\_M* : afecta els mòduls de la fase [pfc] identificats com a màster.

*A10\*\** : afecta els mòduls que estan connectats al Nord però no a l'Oest.

*T0,1, fw\_\_A* : afecta els mòduls que tenen el veí del Nord en estat de fw actiu i en mode d'espera.

*W0,1, C024 0000* : afecta els mòduls que tenen el veí del Nord encara sense marcar.

- **Postcondicions:**

*MN LStrp* : missatge de "Leader Strip" al Nord.

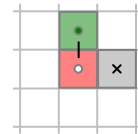
## Annex C - Regles de la fase de reconfiguració a la forma canònica

---

<b>Regla:</b>	Receive Mark Leaf Strip
<b>Prioritat:</b>	8020
<b>Precondicions:</b>	<i>Sfw***</i> <i>A0001</i> <i>T0,-1, pfc_M</i> <i>MS LStrp</i>
<b>Postcondicions:</b>	<i>C024 + 0000 0001</i>

### - Regla: Receive Mark Leaf Strip

Aquesta regla s'aplica als mòduls que reben un missatge de "Leader Strip" del màster.



### - Prioritat: 8020

La prioritat és més gran que la de les regles de moviment per evitar que el mòdul es mogui abans de gestionar la marca de líder de la tira.

### - Precondicions:

*Sfw\*\*\** : afecta els mòduls de la fase [fw].

*A0001* : afecta els mòduls que estan connectats només al Sud.

*T0,-1, pfc\_M* : afecta els mòduls que tenen el màster al Sud.

*MS LStrp* : afecta els mòduls que reben un missatge de "Leader Strip" del Sud.

### - Postcondicions:

*C024 + 0000 0001* : el mòdul es marca com a líder de la tira.

## Annex D

### *Regles de la fase de construcció de la forma final*

En aquest apartat es detallen les regles utilitzades en cada grup de la fase de construcció de la forma final [bfs] (*Build final shape*). Les regles descrites són les encarregades de generar les dades de la forma final que necessita el sistema per tal que el màster les pugui comunicar als mòduls a mida que inicien el procés de construcció de la forma final. Les regles que realitzen la construcció de la forma final pròpiament dita no estan disponibles. L'objectiu, l'estratègia i una visió general d'aquest conjunt de regles es descriuen al Capítol 7.

Totes les regles d'aquesta fase inclouen "[R]" (*Reverse*) a l'inici del seu nom per diferenciar-les de les normals, i tots els estats tenen com a primer caràcter una 'R'.

#### **D.1 Grup: "reverse rules"**

Aquest grup està format per les mateixes regles que els seus grups anàlegs de la versió normal (no invertida):

- 124 regles de la fase [ini] ("*master*" & "*holes*").
- 4 regles de la fase [bst].
- 105 regles de la fase [bpf]:
  - 3 regles del grup 1 ("*start*")
  - 24 regles del grup 2 ("*first go*")
  - 4 regles del grup 3 ("*first wait*")
  - 24 regles del grup 4 ("*wait-continue*")
  - 8 regles del grup 5 ("*continue*")
  - 12 regles del grup 6 ("*last*")
  - 2 regles del grup 7 ("*last father*")
  - 28 regles del grup 8 ("*first go adapt*" i "*first wait adapt*")

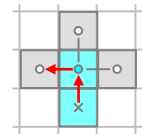
Aquestes regles tenen les mateixes condicions (excepte les que determinen la direcció, que estan invertides) i realitzen la mateixa funció però amb els resultats invertits. La diferència (a part del nom i els estats) és que l'ordre que segueixen és l'invers. Els mòduls ara segueixen la regla de la mà esquerra. Per exemple, si en les regles normals el camí preferent era cap a l'Est, ara és cap a l'Oest. Les regles són afectades per aquestes modificacions són les de la fase [bpf].

En aquest grup només mostrem un exemple com a referència. La regla mostrada és la inversa de la "Spread potential function first from S to E new branch". (Ressaltem les diferències de la versió normal a la "reverse" en l'exemple).

## Annex D - Regles de la fase de construcció de la forma final

Regla:	[R] Spread potential function first from S to <b>W</b> new branch
Prioritat:	10000
Precondicions:	<p><b>S</b> <b>R</b> <b>p</b> <b>f</b> <b>**</b></p> <p><b>S</b> <b>**</b> <b>*</b></p> <p>=#S05 0001</p> <p><b>A</b> <b>*</b> <b>1</b> <b>*</b> <b>1</b></p> <p><b>T</b> <b>-1,0</b>, <b>R</b> <b>p</b> <b>f</b> <b>_</b> <b>*</b></p> <p>!(</p> <p>!(<b>A1101</b> !T0,1, <b>R</b> <b>n</b> <b>i</b> <b>**</b> !T0,1, <b>R</b> <b>s</b> <b>t</b> <b>**</b> )</p> <p>!(<b>A0111</b> !T<b>1,0</b>, <b>R</b> <b>n</b> <b>i</b> <b>**</b> !T<b>1,0</b>, <b>R</b> <b>s</b> <b>t</b> <b>**</b> )</p> <p>!(<b>A1111</b> !T0,1, <b>R</b> <b>n</b> <b>i</b> <b>**</b> !T0,1, <b>R</b> <b>s</b> <b>t</b> <b>**</b> !T<b>1,0</b>, <b>R</b> <b>n</b> <b>i</b> <b>**</b> !T<b>1,0</b>, <b>R</b> <b>s</b> <b>t</b> <b>**</b> )</p> <p>)</p>
Postcondicions:	<p>C000 + #S01 0001</p> <p>C002 + 0000 0004</p> <p>C003 + #S02 0000</p> <p>#<b>W</b>05 + 0000 0001</p> <p>#<b>W</b>01 + C000 0000</p> <p>#<b>W</b>02 + C000 0001</p> <p><b>S</b> <b>**</b> <b>m</b> <b>*</b></p>

- **Regla:** [R] Spread potential function first from S to **W** new branch  
Aquesta regla s'aplica als mòduls als quals arriba per primer cop el missatge de funció potencial (encara no estan marcats) pel Sud i l'han de propagar per l'**Oest**, que forma part d'una nova branca perquè es troba en una bifurcació de diversos mòduls connectats.



- **Prioritat:** 10000

La prioritat és la mateixa que la de les fases [ini - **reverse**] i [bst - **reverse**] per poder treballar en paral·lel.

- **Precondicions:**

**S** **R** **p** **f** **\*\*** : afecta els mòduls de la fase [bpf - **reverse**]

**S** **\*\*** **\*** : afecta els mòduls que no han estat marcats.

=#S05 0001 : afecta els mòduls que reben un missatge de "marca" del Sud.

**A** **\*** **1** **\*** **1** : afecta els mòduls que estan connectats al Sud i a l'**Oest**.

**T** **-1,0**, **R** **p** **f** **\_** **\*** : afecta els mòduls que tenen el veí de l'**Oest** en estat de bpf\_ - **reverse**.

!( !(**A1101** !T0,1, **R** **n** **i** **\*\*** !T0,1, **R** **s** **t** **\*\*** ) !(**A0111** !T**1,0**, **R** **n** **i** **\*\*** !T**1,0**, **R** **s** **t** **\*\*** ) !(**A1111** !T0,1, **R** **n** **i** **\*\*** !T0,1, **R** **s** **t** **\*\*** !T**1,0**, **R** **n** **i** **\*\*** !T**1,0**, **R** **s** **t** **\*\*** )) : afecta els mòduls que estan connectats a altres mòduls a part de Sud i **Oest**, i tots ells han superat ja les fases de [ini - **reverse**] i [bst - **reverse**].

- **Postcondicions:**

C000 + #S01 0001 :  $V_{pot} = V_{pot}(S) + 1$

C002 + 0000 0004 : Pare = Sud

C003 + #S02 0000 : Min = Min(S)

#**W**05 + 0000 0001 : missatge de "marca" a l'**Oest**.

#**W**01 + C000 0000 : missatge pel canal 1 a l'**Oest**: ( $V_{pot} = V_{pot}$ ).

#**W**02 + C000 0001 : missatge pel canal 2 a l'**Oest**: ( $Min = V_{pot} + 1$ ).

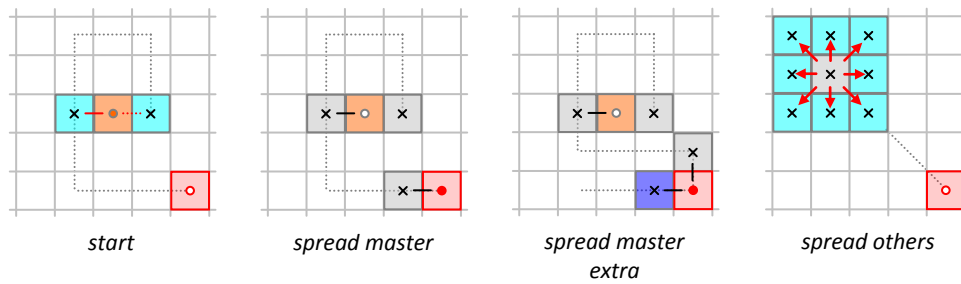
**S** **\*\*** **m** **\*** : canvia el seu estat a "marcat".

## D.2 Grup: "rollback"

Aquest grup està format per 4 regles: 1 regla que inicia el rollback i 3 que el propaguen.

En aquest grup hem implementat les regles següents:

### - Rollback

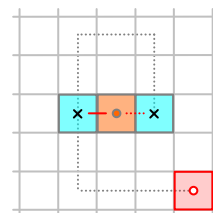


## Annex D - Regles de la fase de construcció de la forma final

<b>Regla:</b>	<i>[R] Start rollback (special hole detected)</i>
<b>Prioritat:</b>	<i>10001</i>
<b>Precondicions:</b>	
<i>SRpf**</i> <i>S*** *</i> <i>S****H</i> <i>=#E05 0001</i> <i>T-1,0, Rpfm*</i> <i>V-1,0, C000 #E01</i>	
<b>Postcondicions:</b>	
<i>A*10*</i> <i>#*Q5 + 0000 0000</i>	

- Regla: [R] Start rollback (special hole detected)

*Aquesta regla s'aplica als mòduls líder d'un forat que han de canviar les seves connexions per tractar-se d'un forat especial (special hole).*



- Prioritat: 10001

*La prioritat és superior a la resta de regles perquè el reseteig de les dades té preferència sobre la resta de processos.*

- Precondicions:

SRpf\*\* : *afecta els mòduls de la fase [bpf - reverse]*

S\*\*\* \* : afecta els mòduls que no han estat marcats.

**S\*\*\*\*H** : *afecta els mòduls identificats com a líders de forat.*

=#E05 0001 : *afecta els mòduls que reben un missatge de "marca" de l'Est.*

T-1,0, Rpfm\* : afecta els mòduls que tenen el veí de l'Oest en estat de bpfm reverse.

V-1,0, C000 #E01 : *afecta els mòduls que tindrien un valor potencial més gran que el veí de l'Oest.*

- Postcondiciones:

$A^*10^*$  : es desconnecta de l'Est i es connecta a l'Oest.

#\*05 + 0000 0000 : missatge de “reset” en totes direccions.



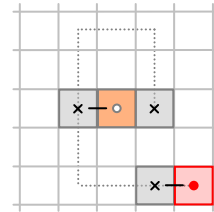
Regla:	[R] Spread rollback master			
Prioritat:	10001			
Precondicions:				
S****M !( !SRpfm* !SRfc_* !SRgi_* ) =#*05 0000				
Postcondicions:				
C000 + 0000 0000	C001 + 0000 0000	C002 + 0000 0000	C003 + 0000 0000	C004 + 0000 0000
C005 + 0000 0000	C006 + 0000 0000	C007 + 0000 0000	C008 + 0000 0000	C009 + 0000 0000
C010 + 0000 0000	C011 + 0000 0000	C012 + 0000 0000	C013 + 0000 0000	C014 + 0000 0000
C015 + 0000 0000	C016 + 0000 0000	C017 + 0000 0000	C018 + 0000 0000	C019 + 0000 0000
C020 + 0000 0000	C021 + 0000 0000	C022 + 0000 0000	C023 + 0000 0000	C024 + 0000 0000
SRpf_*				

- **Regla:** [R] Spread rollback master

Aquesta regla s'aplica al mòdul màster que rep un senyal de reset.

- **Prioritat:** 10001

La prioritat és superior a la resta de regles perquè el reseteig de les dades té preferència sobre la resta de processos.



- **Precondicions:**

S\*\*\*\*M : afecta els mòduls identificats com a màster.

!( !SRpfm\* !SRfc\_\* !SRgi\_\* ) : afecta els mòduls en estat reverse de bpfm, pfc\_ o Rgi\_.

=#\*05 0000 : afecta els mòduls que reben un missatge de "reset" de qualsevol direcció.

- **Postcondicions:**

C0\*\* + 0000 0000 : neteja tots els comptadors.

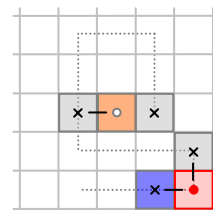
SRpf\_\* : canvia el seu estat a [bpf - reverse] en estat inicial.

## Annex D - Regles de la fase de construcció de la forma final

<b>Regla:</b>	<i>[R] Spread rollback master extra</i>
<b>Prioritat:</b>	<i>10001</i>
<b>Precondicions:</b>	
<i>S****M</i> <i>!( !SRpfm* !SRfc_ * !SRgi_ * )</i> <i>=#N05 0000</i> <i>A11**</i> <i>!W-1,0, C004 0000</i>	
<b>Postcondicions:</b>	
<i>#N05 + 0000 0001</i> <i>#N01 - -1,0, C004 0001</i> <i>#N02 + -1,0, C004 0000</i> <i>S***m*</i>	

- Regla: [R] Spread rollback master extra

*Aquesta regla s'aplica al mòdul màster que rep un senyal de reset del Nord mentre el veí de l'Oest ja disposa del valor màxim.*



- Prioritat: 10001

*La prioritat és superior a la resta de regles perquè el reseteg de les dades té preferència sobre la resta de processos.*

- Precondicions:

**S\*\*\*\*M** : *afecta els mòduls identificats com a màster.*

!( !SRpfm\* !SRfc\_\* !SRgi\_\* ) : afecta els mòduls en estat reverse de bpfm, pfc\_ o Rgi\_.

*=#\*05 0000 : afecta els mòduls que reben un missatge de "reset" del Nord.*

A11\*\* : afecta els mòduls que estan connectats al Nord i a l'Oest.

!W-1,0, C004 0000 : *afecta els mòduls que tenen un veí a l'Oest amb el valor màxim assignat.*

- Postcondiciones:

#N05 + 0000 0001 : missatge de “marca” al Nord.

#N01 - -1,0, C004 0001 : *missatge pel canal 1 al Nord: ( $V_{pot} = V_{pot}(W)$ ).*

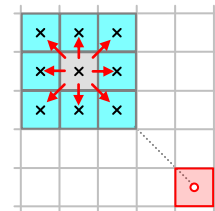
#N02 + -1,0, C004 0000 : *missatge pel canal 2 al Nord: (Min = Vpot(W) + 1).*

$S^{***}m^*$  : canvia el seu estat a "marcat".

Regla:	[R] Spread rollback others			
Prioritat:	10001			
Precondicions:				
!S***M !( !SRpfm* !SRfc_ * !SRgi_ * ) =#*05 0000				
Postcondicions:				
C000 + 0000 0000	C001 + 0000 0000	C002 + 0000 0000	C003 + 0000 0000	C004 + 0000 0000
C005 + 0000 0000	C006 + 0000 0000	C007 + 0000 0000	C008 + 0000 0000	C009 + 0000 0000
C010 + 0000 0000	C011 + 0000 0000	C012 + 0000 0000	C013 + 0000 0000	C014 + 0000 0000
C015 + 0000 0000	C016 + 0000 0000	C017 + 0000 0000	C018 + 0000 0000	C019 + 0000 0000
C020 + 0000 0000	C021 + 0000 0000	C022 + 0000 0000	C023 + 0000 0000	C024 + 0000 0000
#*05 + 0000 0000 SRpf_*				

- **Regla:** [R] Spread rollback others

Aquesta regla s'aplica als mòduls que no són màster que reben un senyal de reset i el propaguen per tot arreu.



- **Prioritat:** 10001

La prioritat és superior a la resta de regles perquè el reseteig de les dades té preferència sobre la resta de processos.

- **Precondicions:**

IS\*\*\*M : afecta els mòduls que no són identificats com a màster.

!( !SRpfm\* !SRfc\_\* !SRgi\_\* ) : afecta els mòduls en estat reverse de bpfm, pfc\_ o Rgi\_.

=#\*05 0000 : afecta els mòduls que reben un missatge de "reset" de qualsevol direcció.

- **Postcondicions:**

C0\*\* + 0000 0000 : neteja tots els comptadors.

#\*05 + 0000 0000 : missatge de "reset" en totes direccions.

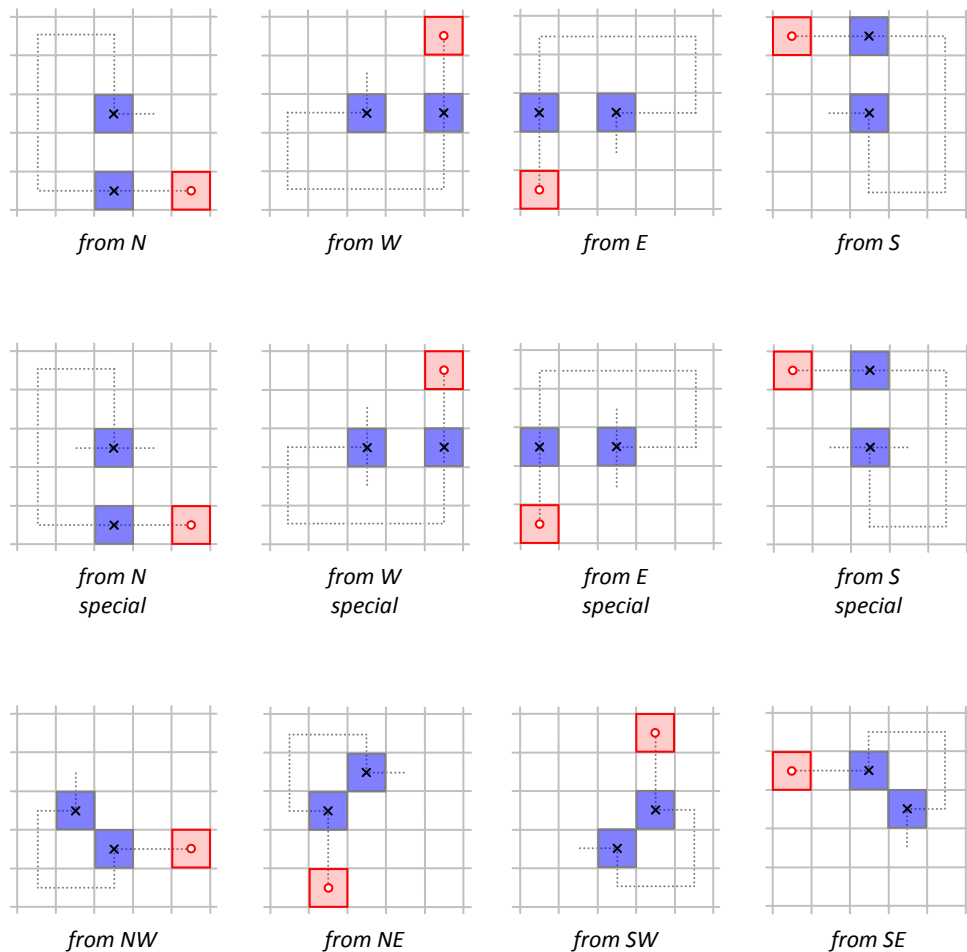
SRpf\_\* : canvia el seu estat a [bpf - reverse] en estat inicial.

### D.3 Grup: “get info”

Aquest grup està format per 15 regles subdividides en 3 subgrups. El primer subgrup consta de 12 regles per capturar les dades dels mòduls que generen el coll d'ampolla, 8 regles per als mòduls que es troben de front (4 per bifurcacions normals, 4 per bifurcacions especials) i 4 per als mòduls que estan en cantonada. El segon subgrup té 2 regles, una per cada tipus de responsable de forat depenent de la forma que té el tancament del forat. El tercer subgrup és d'una única regla que canvia d'estat si ha capturat tota la informació possible del seu entorn.

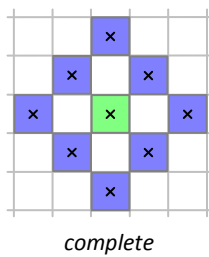
En aquest grup hem implementat les regles següents:

#### - Get bottleneck info

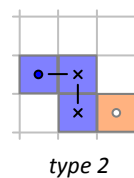
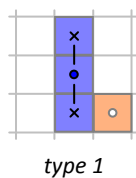


Nota: la posició del màster en aquestes figures no té significació geomètrica, sinó tan sols combinatòria i, per tant, sols contribueix a indicar les posicions relatives dels mòduls al llarg de l'arbre generador del robot.

- Get bottleneck info



- Get hole responsible

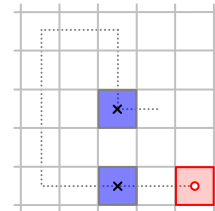


- **Front normal:**

<b>Regla:</b>	[R] Get bottleneck info from N
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
<i>SRfc_*</i> <i>E0,1</i> <i>F0,2</i> <i>!( !T0,2, Rfc_* !T0,2, Rgi_* )</i> <i>!( !T-1,2, Rfc_* !T-1,2, Rgi_* !E-1,2 )</i> <i>=C008 0000</i> <i>!W0,2, C000 C000</i> <i>!( !=C002 0003 !=C002 0004 )</i> <i>!( !(W0,2, C002 0002 !V0,2, C002 0002 ) !( (W0,2, C002 0001 !V0,2, C002 0001 )</i> <i>!( !E-1,2 (W-1,2, C002 0003 !V-1,2, C002 0003 )))</i>	
<b>Postcondicions:</b>	
<i>C008 + 0,2, C000 0000</i> <i>C009 + 0,2, C004 0000</i>	

- **Regla:** [R] Get bottleneck info from N

Aquesta regla s'aplica als mòduls que tenen al Nord el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

*SRfc\_\** : afecta els mòduls de la fase [pfc - reverse].

*E0,1* : afecta els mòduls que tenen un espai buit al Nord.

*F0,2* : afecta els mòduls que tenen un veí al Nord Nord.

*!( !T0,2, Rfc\_\* !T0,2, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Nord Nord està en la fase [pfc - reverse] o [Rgi].

*!( !T-1,2, Rfc\_\* !T-1,2, Rgi\_\* !E-1,2 )* : afecta els mòduls per als quals el seu veí del Nord Nord-Oest no existeix o està en la fase [pfc - reverse] o [Rgi].

*=C008 0000* : afecta els mòduls que encara no han capturat aquesta informació.

*!W0,2, C000 C000* : afecta els mòduls que tenen un veí al Nord Nord amb un valor potencial més gran.

*!( !=C002 0003 !=C002 0004 )* : afecta els mòduls que tenen el pare a l'Est o al Sud.

*!( !(W0,2, C002 0002 !V0,2, C002 0002 ) !( (W0,2, C002 0001 !V0,2, C002 0001 ) !( !E-1,2 (W-1,2, C002 0003 !V-1,2, C002 0003 )))* : identifica les formes de branca que generen el coll d'ampolla "normal".

- **Postcondicions:**

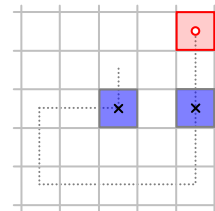
*C008 + 0,2, C000 0000* : Sac N (entrada) = Vpot(N-N)

*C009 + 0,2, C004 0000* : Sac N (salt) = Max (N-N)

<b>Regla:</b>	[R] Get bottleneck info from W
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E-1,0 F-2,0 !( !T-2,0, Rfc_* !T-2,0, Rgi_* ) !( !T-2,-1, Rfc_* !T-2,-1, Rgi_* !E-2,-1 ) =C010 0000 !W-2,0, C000 C000 !( !=C002 0001 !=C002 0003 ) !( !(W-2,0, C002 0004 !V-2,0, C002 0004 ) !( (W-2,0, C002 0002 !V-2,0, C002 0002 ) !( !E-2,-1 (W-2,-1, C002 0001 !V-2,-1, C002 0001 ) ) ) ) )	
<b>Postcondicions:</b>	
C010 + -2,0, C000 0000 C011 + -2,0, C004 0000	

- **Regla:** [R] Get bottleneck info from W

Aquesta regla s'aplica als mòduls que tenen a l'Oest el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E-1,0 : afecta els mòduls que tenen un espai buit a l'Oest.

F-2,0 : afecta els mòduls que tenen un veí a l'Oest Oest.

!( !T-2,0, Rfc\_\* !T-2,0, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Oest Oest està en la fase [pfc - reverse] o [Rgi].

!( !T-2,-1, Rfc\_\* !T-2,-1, Rgi\_\* !E-2,-1 ) : afecta els mòduls per als quals el seu veí de l'Oest Sud-Oest no existeix o està en la fase [pfc - reverse] o [Rgi].

=C010 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W-2,0, C000 C000 : afecta els mòduls que tenen un veí a l'Oest Oest amb un valor potencial més gran.

!( !=C002 0001 !=C002 0003 ) : afecta els mòduls que tenen el pare al Nord o a l'Est.

!( !(W-2,0, C002 0004 !V-2,0, C002 0004 ) !( (W-2,0, C002 0002 !V-2,0, C002 0002 ) !( !E-2,-1 (W-2,-1, C002 0001 !V-2,-1, C002 0001 ) ) ) ) ) : identifica les formes de branca que generen el coll d'ampolla "normal".

- **Postcondicions:**

C010 + -2,0, C000 0000 : Sac W (entrada) = Vpot(W-W)

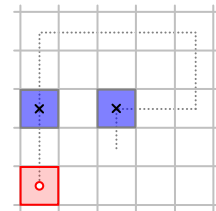
C011 + -2,0, C004 0000 : Sac W (salt) = Max (W-W)

## Annex D - Regles de la fase de construcció de la forma final

<b>Regla:</b>	[R] Get bottleneck info from E
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E1,0 F2,0 !( !T2,0, Rfc_* !T2,0, Rgi_* ) !( !T2,1, Rfc_* !T2,1, Rgi_* !E2,1 ) =C012 0000 !W2,0, C000 C000 !( !=C002 0004 !=C002 0002 ) !( !(W2,0, C002 0001 !V2,0, C002 0001 ) !( (W2,0, C002 0003 !V2,0, C002 0003 ) !( !E2,1 (W2,1, C002 0004 !V2,1, C002 0004 ))) )	
<b>Postcondicions:</b>	
C012 + 2,0, C000 0000 C013 + 2,0, C004 0000	

- **Regla:** [R] Get bottleneck info from E

Aquesta regla s'aplica als mòduls que tenen a l'Est el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E1,0 : afecta els mòduls que tenen un espai buit a l'Est.

F2,0 : afecta els mòduls que tenen un veí a l'Est Est.

!( !T2,0, Rfc\_\* !T2,0, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Est Est està en la fase [pfc - reverse] o [Rgi].

!( !T2,1, Rfc\_\* !T2,1, Rgi\_\* !E2,1 ) : afecta els mòduls per als quals el seu veí de l'Est Nord-Est no existeix o està en la fase [pfc - reverse] o [Rgi].

=C012 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W2,0, C000 C000 : afecta els mòduls que tenen un veí a l'Est Est amb un valor potencial més gran.

!( !=C002 0004 !=C002 0002 ) : afecta els mòduls que tenen el pare al Sud o a l'Oest.

!( !(W2,0, C002 0001 !V2,0, C002 0001 ) !( (W2,0, C002 0003 !V2,0, C002 0003 ) !( !E2,1 (W2,1, C002 0004 !V2,1, C002 0004 ))) ) : identifica les formes de branca que generen el coll d'ampolla "normal".

- **Postcondicions:**

C012 + 2,0, C000 0000 : Sac E (entrada) = Vpot(E-E)

C013 + 2,0, C004 0000 : Sac E (salt) = Max (E-E)



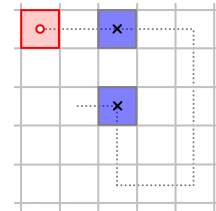
<b>Regla:</b>	[R] Get bottleneck info from S
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E0,-1 F0,-2 !( !T0,-2, Rfc_* !T0,-2, Rgi_* ) !( !T1,-2, Rfc_* !T1,-2, Rgi_* !E1,-2 ) =C014 0000 !W0,-2, C000 C000 !( !=C002 0002 !=C002 0001 ) !( !(W0,-2, C002 0003 !V0,-2, C002 0003 ) !( (W0,-2, C002 0004 !V0,-2, C002 0004 ) !( !E1,-2 (W1,-2, C002 0002 !V1,-2, C002 0002 ) ) ) )	
<b>Postcondicions:</b>	
C014 + 0,-2, C000 0000 C015 + 0,-2, C004 0000	

- **Regla:** [R] Get bottleneck info from S

Aquesta regla s'aplica als mòduls que tenen al Sud el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.

- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.



- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E0,-1 : afecta els mòduls que tenen un espai buit al Sud.

F0,-2 : afecta els mòduls que tenen un veí al Sud Sud.

!( !T0,-2, Rfc\_\* !T0,-2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Sud Sud està en la fase [pfc - reverse] o [Rgi].

!( !T1,-2, Rfc\_\* !T1,-2, Rgi\_\* !E1,-2 ) : afecta els mòduls per als quals el seu veí del Sud Sud-Est no existeix o està en la fase [pfc - reverse] o [Rgi].

=C014 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W0,-2, C000 C000 : afecta els mòduls que tenen un veí al Sud Sud amb un valor potencial més gran.

!( !=C002 0002 !=C002 0001 ) : afecta els mòduls que tenen el pare a l'Oest o al Nord.

!( !(W0,-2, C002 0003 !V0,-2, C002 0003 ) !( (W0,-2, C002 0004 !V0,-2, C002 0004 ) !( !E1,-2 (W1,-2, C002 0002 !V1,-2, C002 0002 ) ) ) ) : identifica les formes de branca que generen el coll d'ampolla "normal".

- **Postcondicions:**

C014 + 0,-2, C000 0000 : Sac S (entrada) = Vpot(S-S)

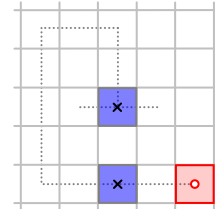
C015 + 0,-2, C004 0000 : Sac S (salt) = Max (S-S)

- Front especial:

<b>Regla:</b>	[R] Get bottleneck info from N special
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E0,1 F0,2 !( !T0,2, Rfc_* !T0,2, Rgi_* ) F-1,2 !( !T-1,2, Rfc_* !T-1,2, Rgi_* ) F1,2 !( !T1,2, Rfc_* !T1,2, Rgi_* ) =C008 0000 !W0,2, C000 C000 !( !=C002 0003 !=C002 0004 ) W0,2, C002 0001 !V0,2, C002 0001 W-1,2, C002 0003 !V-1,2, C002 0003 W1,2, C002 0002 !V1,2, C002 0002	
<b>Postcondicions:</b>	
C008 + 0,2, C000 0000 C009 + 1,2, C004 0000	

- Regla: [R] Get bottleneck info from N special

Aquesta regla s'aplica als mòduls que tenen al Nord el mòdul que genera el coll d'ampolla amb dues subbranques, una interior i una altra exterior al cul de sac, i determina quins mòduls hauran de saltar.



- Prioritat: 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- Precondicions:

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E0,1 : afecta els mòduls que tenen un espai buit al Nord.

F0,2 : afecta els mòduls que tenen un veí al Nord Nord.

!( !T0,2, Rfc\_\* !T0,2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Nord Nord està en la fase [pfc - reverse] o [Rgi].

F-1,2 !( !T-1,2, Rfc\_\* !T-1,2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Nord Nord-Oest està en la fase [pfc - reverse] o [Rgi].

F1,2 !( !T1,2, Rfc\_\* !T1,2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Nord Nord-Est està en la fase [pfc - reverse] o [Rgi].

=C008 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W0,2, C000 C000 : afecta els mòduls que tenen un veí al Nord Nord amb un valor potencial més gran.

!( !=C002 0003 !=C002 0004 ) : afecta els mòduls que tenen el pare a l'Est o al Sud.

W0,2, C002 0001 !V0,2, C002 0001 W-1,2, C002 0003 !V-1,2, C002 0003 W1,2, C002 0002 !V1,2, C002 0002 : identifica la forma de branca "especial".

- Postcondicions:

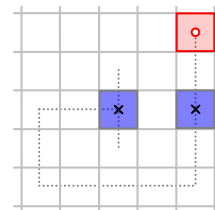
C008 + 0,2, C000 0000 : Sac N (entrada) = Vpot(N-N)

C009 + 1,2, C004 0000 : Sac N (salt) = Max (N-NE)

<b>Regla:</b>	[R] Get bottleneck info from W special
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E-1,0 F-2,0 !( !T-2,0, Rfc_* !T-2,0, Rgi_* ) F-2,-1 !( !T-2,-1, Rfc_* !T-2,-1, Rgi_* ) F-2,1 !( !T-2,1, Rfc_* !T-2,1, Rgi_* ) =C010 0000 !W-2,0, C000 C000 !( !=C002 0001 !=C002 0003 ) W-2,0, C002 0002 !V-2,0, C002 0002 W-2,-1, C002 0001 !V-2,-1, C002 0001 W-2,1, C002 0004 !V-2,1, C002 0004	
<b>Postcondicions:</b>	
C010 + -2,0, C000 0000 C011 + -2,1, C004 0000	

- **Regla:** [R] Get bottleneck info from W special

Aquesta regla s'aplica als mòduls que tenen a l'Oest el mòdul que genera el coll d'ampolla amb dues subbranques, una interior i una altra exterior al cul de sac, i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E-1,0 : afecta els mòduls que tenen un espai buit a l'Oest.

F-2,0 : afecta els mòduls que tenen un veí a l'Oest Oest.

!( !T-2,0, Rfc\_\* !T-2,0, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Oest Oest està en la fase [pfc - reverse] o [Rgi].

F-2,-1 !( !T-2,-1, Rfc\_\* !T-2,-1, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Oest Sud-Oest està en la fase [pfc - reverse] o [Rgi].

F-2,1 !( !T-2,1, Rfc\_\* !T-2,1, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Oest Nord-Oest està en la fase [pfc - reverse] o [Rgi].

=C010 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W-2,0, C000 C000 : afecta els mòduls que tenen un veí a l'Oest Oest amb un valor potencial més gran.

!( !=C002 0001 !=C002 0003 ) : afecta els mòduls que tenen el pare al Nord o a l'Est.

W-2,0, C002 0002 !V-2,0, C002 0002 W-2,-1, C002 0001 !V-2,-1, C002 0001 W-2,1, C002 0004 !V-2,1, C002 0004 : identifica la forma de branca "especial".

- **Postcondicions:**

C010 + -2,0, C000 0000 : Sac W (entrada) = Vpot(W-W)

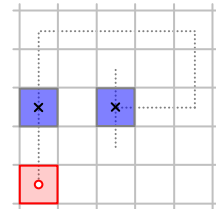
C011 + -2,1, C004 0000 : Sac W (salt) = Max (W-NW)

## Annex D - Regles de la fase de construcció de la forma final

<b>Regla:</b>	[R] Get bottleneck info from E special
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E1,0 F2,0 !( !T2,0, Rfc_* !T2,0, Rgi_* ) F2,1 !( !T2,1, Rfc_* !T2,1, Rgi_* ) F2,-1 !( !T2,-1, Rfc_* !T2,-1, Rgi_* ) =C012 0000 !W2,0, C000 C000 !( !=C002 0004 !=C002 0002 ) W2,0, C002 0003 !V2,0, C002 0003 W2,1, C002 0004 !V2,1, C002 0004 W2,-1, C002 0001 !V2,-1, C002 0001	
<b>Postcondicions:</b>	
C012 + 2,0, C000 0000 C013 + 2,-1, C004 0000	

- **Regla:** [R] Get bottleneck info from E special

Aquesta regla s'aplica als mòduls que tenen a l'Est el mòdul que genera el coll d'ampolla amb dues subbranques, una interior i una altra exterior al cul de sac, i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E1,0 : afecta els mòduls que tenen un espai buit a l'Est.

F2,0 : afecta els mòduls que tenen un veí a l'Est Est.

!( !T2,0, Rfc\_\* !T2,0, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Est Est està en la fase [pfc - reverse] o [Rgi].

F2,1 !( !T2,1, Rfc\_\* !T2,1, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Est Nord-Est està en la fase [pfc - reverse] o [Rgi].

F2,-1 !( !T2,-1, Rfc\_\* !T2,-1, Rgi\_\* ) : afecta els mòduls per als quals el seu veí de l'Est Sud-Est està en la fase [pfc - reverse] o [Rgi].

=C012 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W2,0, C000 C000 : afecta els mòduls que tenen un veí a l'Est Est amb un valor potencial més gran.

!( !=C002 0004 !=C002 0002 ) : afecta els mòduls que tenen el pare al Sud o a l'Oest.

W2,0, C002 0003 !V2,0, C002 0003 W2,1, C002 0004 !V2,1, C002 0004 W2,-1, C002 0001 !V2,-1, C002 0001 : identifica la forma de branca "especial".

- **Postcondicions:**

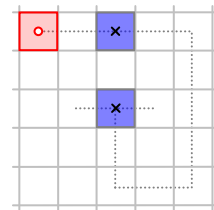
C012 + 2,0, C000 0000 : Sac E (entrada) = Vpot(E-E)

C013 + 2,-1, C004 0000 : Sac E (salt) = Max (E-SE)

<b>Regla:</b>	[R] Get bottleneck info from S special
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E0,-1 F0,-2 !( !T0,-2, Rfc_* !T0,-2, Rgi_* ) F1,-2 !( !T1,-2, Rfc_* !T1,-2, Rgi_* ) F-1,-2 !( !T-1,-2, Rfc_* !T-1,-2, Rgi_* ) =C014 0000 !W0,-2, C000 C000 !( !=C002 0002 !=C002 0001 ) W0,-2, C002 0004 !V0,-2, C002 0004 W1,-2, C002 0002 !V1,-2, C002 0002 W-1,-2, C002 0003 !V-1,-2, C002 0003	
<b>Postcondicions:</b>	
C014 + 0,-2, C000 0000 C015 + -1,-2, C004 0000	

- **Regla:** [R] Get bottleneck info from S special

Aquesta regla s'aplica als mòduls que tenen al Sud el mòdul que genera el coll d'ampolla amb dues subbranques, una interior i una altra exterior al cul de sac, i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E0,-1 : afecta els mòduls que tenen un espai buit al Sud.

F0,-2 : afecta els mòduls que tenen un veí al Sud Sud.

!( !T0,-2, Rfc\_\* !T0,-2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Sud Sud està en la fase [pfc - reverse] o [Rgi].

F1,-2 !( !T1,-2, Rfc\_\* !T1,-2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Sud Sud-Est està en la fase [pfc - reverse] o [Rgi].

F-1,-2 !( !T-1,-2, Rfc\_\* !T-1,-2, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Sud Sud-Oest està en la fase [pfc - reverse] o [Rgi].

=C014 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W0,-2, C000 C000 : afecta els mòduls que tenen un veí al Sud Sud amb un valor potencial més gran.

!( !=C002 0002 !=C002 0001 ) : afecta els mòduls que tenen el pare a l'Oest o al Nord.

W0,-2, C002 0004 !V0,-2, C002 0004 W1,-2, C002 0002 !V1,-2, C002 0002 W-1,-2, C002 0003 !V-1,-2, C002 0003 : identifica la forma de branca "especial".

- **Postcondicions:**

C014 + 0,-2, C000 0000 : Sac S (entrada) = Vpot(S-S)

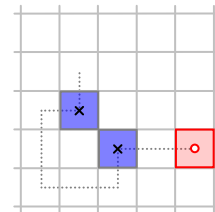
C015 + -1,-2, C004 0000 : Sac S (salt) = Max (S-SW)

- **Cantonada:**

<b>Regla:</b>	[R] Get bottleneck info from NW
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
<i>SRfc_*</i> <i>E0,1</i> <i>E-1,0</i> <i>F-1,1</i> <i>!( !T-1,1, Rfc_* !T-1,1, Rgi_* )</i> <i>=C016 0000</i> <i>!W-1,1, C000 C000</i> <i>=C002 0003</i> <i>W-1,1, C002 0002 !V-1,1, C002 0002</i>	
<b>Postcondicions:</b>	
<i>C016 + -1,1, C000 0000</i> <i>C017 + -1,1, C004 0000</i>	

- **Regla:** [R] Get bottleneck info from NW

Aquesta regla s'aplica als mòduls que tenen al Nord-Oest el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

*SRfc\_\** : afecta els mòduls de la fase [pfc - reverse].

*E0,1* : afecta els mòduls que tenen un espai buit al Nord.

*E-1,0* : afecta els mòduls que tenen un espai buit a l'Oest.

*F-1,1* : afecta els mòduls que tenen un veí al Nord-Oest.

*!( !T-1,1, Rfc\_\* !T-1,1, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Nord-Oest està en la fase [pfc - reverse] o [Rgi].

*=C016 0000* : afecta els mòduls que encara no han capturat aquesta informació.

*!W-1,1, C000 C000* : afecta els mòduls que tenen un veí al Nord-Oest amb un valor potencial més gran.

*=C002 0003* : afecta els mòduls que tenen el pare a l'Est.

*W-1,1, C002 0002 !V-1,1, C002 0002* : identifica la forma de branca que genera el coll d'ampolla.

- **Postcondicions:**

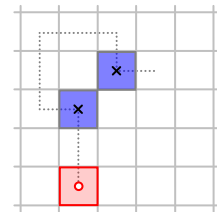
*C016 + -1,1, C000 0000* : *Sac NW (entrada) = Vpot(NW)*

*C017 + -1,1, C004 0000* : *Sac NW (salt) = Max (NW)*

<b>Regla:</b>	[R] Get bottleneck info from NE
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E0,1 E1,0 F1,1 !( !T1,1, Rfc_* !T1,1, Rgi_* ) =C018 0000 !W1,1, C000 C000 =C002 0004 W1,1, C002 0001 !V1,1, C002 0001	
<b>Postcondicions:</b>	
C018 + 1,1, C000 0000 C019 + 1,1, C004 0000	

- **Regla:** [R] Get bottleneck info from NE

Aquesta regla s'aplica als mòduls que tenen al Nord-Est el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E0,1 : afecta els mòduls que tenen un espai buit al Nord.

E1,0 : afecta els mòduls que tenen un espai buit a l'Est.

F1,1 : afecta els mòduls que tenen un veí al Nord-Est.

!( !T1,1, Rfc\_\* !T1,1, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Nord-Est està en la fase [pfc - reverse] o [Rgi].

=C018 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W1,1, C000 C000 : afecta els mòduls que tenen un veí al Nord-Est amb un valor potencial més gran.

=C002 0004 : afecta els mòduls que tenen el pare al Sud.

W1,1, C002 0001 !V1,1, C002 0001 : identifica la forma de branca que genera el coll d'ampolla.

- **Postcondicions:**

C018 + 1,1, C000 0000 : Sac NE (entrada) = Vpot(NE)

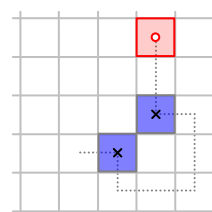
C019 + 1,1, C004 0000 : Sac NE (salt) = Max (NE)

## Annex D - Regles de la fase de construcció de la forma final

<b>Regla:</b>	[R] Get bottleneck info from SW
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
<i>SRfc_*</i> <i>E0,-1</i> <i>E-1,0</i> <i>F-1,-1</i> <i>!( !T-1,-1, Rfc_* !T-1,-1, Rgi_* )</i> <i>=C020 0000</i> <i>!W-1,-1, C000 C000</i> <i>=C002 0001</i> <i>W-1,-1, C002 0004 !V-1,-1, C002 0004</i>	
<b>Postcondicions:</b>	
<i>C020 + -1,-1, C000 0000</i> <i>C021 + -1,-1, C004 0000</i>	

- **Regla:** [R] Get bottleneck info from SW

Aquesta regla s'aplica als mòduls que tenen al Sud-Oest el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

*SRfc\_\** : afecta els mòduls de la fase [pfc - reverse].

*E0,-1* : afecta els mòduls que tenen un espai buit al Sud.

*E-1,0* : afecta els mòduls que tenen un espai buit a l'Oest.

*F-1,-1* : afecta els mòduls que tenen un veí al Sud-Oest.

*!( !T-1,-1, Rfc\_\* !T-1,-1, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Sud-Oest està en la fase [pfc - reverse] o [Rgi].

*=C020 0000* : afecta els mòduls que encara no han capturat aquesta informació.

*!W-1,-1, C000 C000* : afecta els mòduls que tenen un veí al Sud-Oest amb un valor potencial més gran.

*=C002 0001* : afecta els mòduls que tenen el pare al Nord.

*W-1,-1, C002 0004 !V-1,-1, C002 0004* : identifica la forma de branca que genera el coll d'ampolla.

- **Postcondicions:**

*C020 + -1,-1, C000 0000* : Sac SW (entrada) = Vpot(SW)

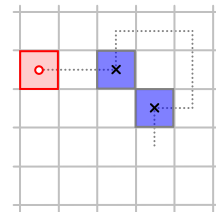
*C021 + -1,-1, C004 0000* : Sac SW (salt) = Max (SW)



<b>Regla:</b>	[R] Get bottleneck info from SE
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* E0,-1 E1,0 F1,-1 !( !T1,-1, Rfc_* !T1,-1, Rgi_* ) =C022 0000 !W1,-1, C000 C000 =C002 0002 W1,-1, C002 0003 !V1,-1, C002 0003	
<b>Postcondicions:</b>	
C022 + 1,-1, C000 0000 C023 + 1,-1, C004 0000	

- **Regla:** [R] Get bottleneck info from SE

Aquesta regla s'aplica als mòduls que tenen al Sud-Est el mòdul que genera el coll d'ampolla i determina quins mòduls hauran de saltar.



- **Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

- **Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

E0,-1 : afecta els mòduls que tenen un espai buit al Sud.

E1,0 : afecta els mòduls que tenen un espai buit a l'Est.

F1,-1 : afecta els mòduls que tenen un veí al Sud-Est.

!( !T1,-1, Rfc\_\* !T1,-1, Rgi\_\* ) : afecta els mòduls per als quals el seu veí del Sud-Est està en la fase [pfc - reverse] o [Rgi].

=C022 0000 : afecta els mòduls que encara no han capturat aquesta informació.

!W1,-1, C000 C000 : afecta els mòduls que tenen un veí al Sud-Est amb un valor potencial més gran.

=C002 0002 : afecta els mòduls que tenen el pare a l'Oest.

W1,-1, C002 0003 !V1,-1, C002 0003 : identifica la forma de branca que genera el coll d'ampolla.

- **Postcondicions:**

C022 + 1,-1, C000 0000 : Sac SE (entrada) = Vpot(SE)

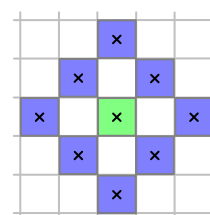
C023 + 1,-1, C004 0000 : Sac SE (salt) = Max (SE)

- **Complete:**

<b>Regla:</b>	[R] Get bottleneck info complete
<b>Prioritat:</b>	8999
<b>Precondicions:</b>	
<i>SRfc_*</i> <i>!( !E0,2 !T0,2, Rfc_* !T0,2, Rgi_* )</i> <i>!( !E-2,0 !T-2,0, Rfc_* !T-2,0, Rgi_* )</i> <i>!( !E2,0 !T2,0, Rfc_* !T2,0, Rgi_* )</i> <i>!( !E0,-2 !T0,-2, Rfc_* !T0,-2, Rgi_* )</i> <i>!( !E-1,1 !T-1,1, Rfc_* !T-1,1, Rgi_* )</i> <i>!( !E1,1 !T1,1, Rfc_* !T1,1, Rgi_* )</i> <i>!( !E-1,-1 !T-1,-1, Rfc_* !T-1,-1, Rgi_* )</i> <i>!( !E1,-1 !T1,-1, Rfc_* !T1,-1, Rgi_* )</i>	
<b>Postcondicions:</b>	
<i>SRgi_*</i>	

- **Regla:** [R] Get bottleneck info complete

Aquesta regla s'aplica als mòduls que han capturat la informació de tots els mòduls veïns que poden generar un coll d'ampolla.



- **Prioritat:** 8999

La prioritat és més petita que la de la resta del grup de captura de dades "get info" perquè la verificació de l'obtenció de totes les dades es produeix quan la captura ha finalitzat.

- **Precondicions:**

*SRfc\_\** : afecta els mòduls de la fase [pfc - reverse].

*!( !E0,2 !T0,2, Rfc\_\* !T0,2, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Nord Nord no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E-2,0 !T-2,0, Rfc\_\* !T-2,0, Rgi\_\* )* : afecta els mòduls per als quals el seu veí de l'Oest Oest no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E2,0 !T2,0, Rfc\_\* !T2,0, Rgi\_\* )* : afecta els mòduls per als quals el seu veí de l'Est Est no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E0,-2 !T0,-2, Rfc\_\* !T0,-2, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Sud Sud no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E-1,1 !T-1,1, Rfc\_\* !T-1,1, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Nord-Oest no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E1,1 !T1,1, Rfc\_\* !T1,1, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Nord-Est no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E-1,-1 !T-1,-1, Rfc\_\* !T-1,-1, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Sud-Oest no existeix o està en la fase [pfc - reverse] o [Rgi].

*!( !E1,-1 !T1,-1, Rfc\_\* !T1,-1, Rgi\_\* )* : afecta els mòduls per als quals el seu veí del Sud-Est no existeix o està en la fase [pfc - reverse] o [Rgi].

- **Postcondicions:**

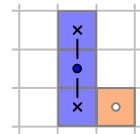
*SRgi\_\** : canvia el seu estat a [Rgi] (Reverse Get Info) "informació disponible".

**- Hole responsible:**

<b>Regla:</b>	[R] Get hole responsible Type 1
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	
SRfc_* F0,-1 F1,-1 T1,-1, ****H =C001 0000 =C002 0001	
<b>Postcondicions:</b>	
C001 + 0,-1, C000 0000	

**- Regla:** [R] Get hole responsible Type 1

Aquesta regla s'aplica als mòduls que són responsables de controlar el mòdul que tanca el forat (el tancament del forat té forma de tipus recte).

**- Prioritat:** 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

**- Precondicions:**

SRfc\_\* : afecta els mòduls de la fase [pfc - reverse].

F0,-1 : afecta els mòduls que tenen un veí al Sud.

F1,-1 : afecta els mòduls que tenen un veí al Sud-Est.

T1,-1, \*\*\*\*H : afecta els mòduls que tenen el líder de forat al Sud-Est.

=C001 0000 : afecta els mòduls que encara no han capturat aquesta informació.

=C002 0001 : afecta els mòduls que tenen el pare al Nord.

**- Postcondicions:**

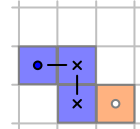
C001 + 0,-1, C000 0000 : Mòdul a controlar = Vpot(S)

## Annex D - Regles de la fase de construcció de la forma final

<b>Regla:</b>	[R] Get hole responsible Type 2
<b>Prioritat:</b>	9000
<b>Precondicions:</b>	<i>SRfc_*</i> <i>F1,0</i> <i>F2,-1</i> <i>T2,-1, ****H</i> <i>=C001 0000</i> <i>W1,0, C002 0002 !V1,0, C002 0002</i>
<b>Postcondicions:</b>	<i>C001 + 1,-1, C000 0000</i>

### - Regla: [R] Get hole responsible Type 2

Aquesta regla s'aplica als mòduls que són responsables de controlar el mòdul que tanca el forat (el tancament del forat té forma de tipus cantonada).



### - Prioritat: 9000

La prioritat és més petita que la del grup "reverse" perquè les dades es capturen quan la resta de processos han finalitzat.

### - Precondicions:

*SRfc\_\** : afecta els mòduls de la fase [pfc - reverse].

*F1,0* : afecta els mòduls que tenen un veí a l'Est.

*F2,-1* : afecta els mòduls que tenen un veí a l'Est Sud-Est.

*T2,-1, \*\*\*\*H* : afecta els mòduls que tenen el líder de forat a l'Est Sud-Est.

*=C001 0000* : afecta els mòduls que encara no han capturat aquesta informació.

*W1,0, C002 0002 !V1,0, C002 0002* : afecta els mòduls que són el pare del seu veí de l'Est.

### - Postcondicions:

*C001 + 1,-1, C000 0000* : Mòdul a controlar = *Vpot(SE)*